402-CD-002-002

EOSDIS Core System Project

System Integration and Test Plan for the ECS Project Volume 2: Release A

Final

February 1995

Hughes Applied Information Systems
Landover, Maryland

This page intentionally left blank.

System Integration and Test Plan for the ECS Project Volume 2: Release A

February 1995

Prepared Under Contract NAS5-60000
CDRL Item 064

SUBMITTED BY

Mark Settle /s/ for	2/7/95
Marshall A. Caplan, Project Manager EOSDIS Core System Project	Date

Hughes Applied Information Systems
Landover, Maryland

402-CD-002-002

This page intentionally left blank.

Preface

This document is a formal contract deliverable with an approval code 1. It requires Government review and approval prior to acceptance and use. Changes to this document also require Government approval prior to acceptance and use. Changes to this document shall be made by document change notice (DCN) or by complete revision.

This document is under ECS Project Configuration Control. Any questions or proposed changes should be addressed to:

Data Management Office
The ECS Project Office
Hughes Applied Information Systems
1616 McCormick Drive
Landover, MD 20785

This page intentionally left blank.

Abstract

The System Integration and Test Plan for the ECS Project Volume 2: Release A (DID 402) contains a plan for testing the TRMM release. The plan was built using a build/thread philosphy. Each thread or build has sequences and test cases defined for testing the objectives/functionality of each software development turnover.

Keywords: *build, thread, dependencies, sequence, test case.*

This page intentionally left blank.

Change Information Page

List of Effective Pages	
Page Number	**Issue**
Title	Final
iii through xii	Final
1-1 through 1-2	Final
2-1 through 2-4	Final
3-1 through 3-14	Final
4-1 and 4-2	Final
5-1 through 5-434	Final
6-1 through 6-2	Final
7-1 through 7-2	Final
8-1 through 8-2	Final
A-1 through A-8	Final
AB-1 through AB-4	Final
GL-1 through GL-20	Final

Document History			
Document Number	**Status/Issue**	**Publication Date**	**CCR Number**
194-402-VE1-001	Original	June 1994	
402-CD-002-002	Review copy	Dec 1994	
402-CD-002-002	Final	Feb 1995	95-0028

This page intentionally left blank.

Contents

Preface

Abstract

1. Introduction

1.1 Identification 1-1
1.2 Scope 1-1
1.3 Purpose and Objectives 1-1
1.4 Status and Schedule 1-1
1.5 Document Organization 1-1

2. Related Documents

2.1 Parent Documents 2-2
2.2 Applicable Documents 2-2
2.3 Information Documents 2-3

3. System Integration and Test Process

3.1 System Build/Thread Test Approach 3-1
3.2 How Testing Is Planned 3-2
3.2.1 Thread/Build Analysis 3-2
3.2.2 Test Plans 3-3
3.2.3 Test Procedures 3-3
3.2.4 Test Conduct 3-4
3.2.5 Consent-To-Ship Review 3-4
3.3 How Testing Is Controlled 3-4
3.4 Roles and Responsibilities 3-6
3.4.1 System Integration and Test 3-6

3.4.2 Configuration Management 3-7
3.4.3 Quality Assurance 3-8
3.5 System I&T in a Multi-Track Environment 3-8
3.6 Integration and Interface Testing 3-9
3.7 Test Tools 3-9
3.8 Tool Kit Testing 3-10
3.8.1 User Interface Tool Kit Testing 3-11
3.8.2 Algorithm Interface (PGS) Tool Kit Testing 3-11
3.8.3 Applications Program Interface (API) Testing 3-12
3.9 Schedules and Dependencies 3-13

4. IR-1 System Thread and Build Plan

5. TRMM (Rel A) System Thread and Build Plan

5.1 TRMM (Rel A) System Threads/Builds 5-4
5.2 TRMM (Rel A) System Build 5-410
5.2.1 TRMM (Rel A) Build (B15) 5-410
5.3 TRMM (Rel A) Interface Testing 5-433
5.3.1 TRMM Mission and Ground System 5-433
5.3.2 Landsat-7 Early Interface 5-433
5.3.3 Early AM-1 Interface 5-433
5.4 TRMM (Rel A) Regression Testing 5-434
5.5 TRMM (Rel A) Site Integration Testing 5-434

Figures

2-1 System Integration & Test Document Relationships 2-1
3-1 Organizational Chart 3-7
3-2 System I&T and Multi-Track Environment 3-8
3-3 PGS Tool Kit Environments 3-12
3-4 PGS SCF Tool Kit Test Environment 3-12
3-5 PGS Testing Environment 3-13

5-1 TRMM (Rel A) System Build/Thread Plan 5-3

Tables

3-1 Configuration Controlled Software Baselines for Test Program 3-7

Abbreviations and Acronyms

Glossary

This page intentionally left blank.

1. Introduction

1.1 Identification

This document is submitted as required by CDRL item 064, DID 402/VE1, whose requirements are specified in this document as a deliverable under the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) contract (NAS5-60000).

1.2 Scope

The ECS System Integration and Test Plan (SITP) for TRMM (Rel A) - Volume 2 describes the process for integrating the major ECS segments and elements, and verifying that the ECS complies with the Functional and Performance Requirements Specification (F&PRS) and the Interface Requirements Documents (IRDs). It identifies a schedule for performing such activities, describes the resource requirements, and the responsible test organizations.

This document reflects the Technical Baseline submitted via contract correspondence number ECS 194-00343.

1.3 Purpose and Objectives

This test plan provides a road map to the system integration and test phase of the verification process by providing a breakdown of the activities to be performed into manageable units called builds and threads. These units are identified and briefly described in the SDR/RIR versions of the document and fully defined in the PDR/IDR updates. The detailed test cases are outlines for the step-by-step test procedures which provide a lower level of detail and give instructions on how to perform the verification of the ECS system.

1.4 Status and Schedule

This version of the ECS TRMM (Rel A) System Integration and Test Plan is due two weeks prior to the Preliminary Design Review (PDR). As an Approval Code 1 document, it requires Government approval prior to acceptance and use. This submittal represents the formal and final version of Volume 2 of the DID 402 document.

Detailed test procedures for the next phase in the system integration test planning arena to support TRMM (Rel A) will be defined in DID 414/VE1, 'ECS System Integration and Test Procedures'.

1.5 Document Organization

The document is organized into five chapters:

Section 1 Introduction, contains the identification, scope, purpose and objectives, status and schedule, and document organization.

Section 2 Related Documents, provides a bibliography of parent, applicable and reference documents for the System Integration and Test Plan.

Section 3 ECS Integration and Test Process, describes the process used to test and verify the ECS.

Section 4 Interim Release 1 System Integration and Testing, defines the corresponding supporting document which contains the IR1 system level thread and build tests.

Section 5 TRMM (Rel A) System Integration and Testing, describes the specific system level thread and build tests, which will be used to verify the functionality of TRMM (Rel A).

Acronyms and a glossary are at the end of the document.

2. Related Documents

Figure 2-1 illustrates the relationships of The System Integration and Test Plan (SITP) to other ECS documents.

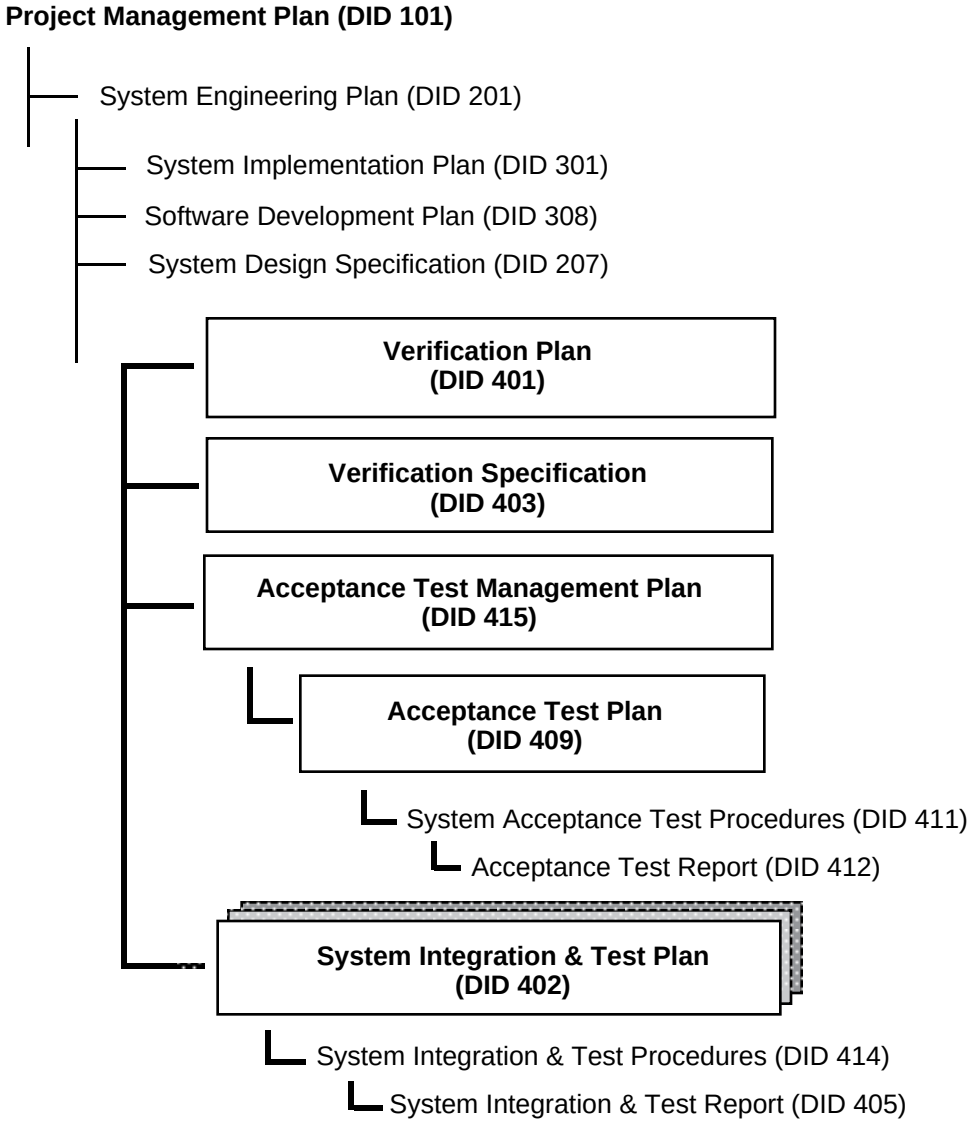

Figure 2-1. System Integration & Test Document Relationships

2.1 Parent Documents

The following documents are the parent documents from which this test plan's scope and content are derived:

194-201-SE1-001	Systems Engineering Plan for the ECS Project
194-207-SE1-001	Systems Design Specification for the ECS Project
194-301-VE1-002	Systems Implementation Plan for the ECS Project
308-CD-001-003	Software Development Plan for the ECS Project
420-05-03	Goddard Space Flight Center, Earth Observing System (EOS) Performance Assurance Requirements for the EOS DIS Core System (ECS)
423-41-02	Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System, Revision A and CH-01
423-41-01	Goddard Space Flight Center, EOSDIS Core System Statement of Work
423-41-03	Goddard Space Flight Center, EOSDIS Core System Contract Data Requirement Document

2.2 Applicable Documents

The following documents are referenced within this plan or are directly applicable, or contain policies or other directive matters that are binding upon the content of this volume:

101-101-MG1-001	Project Management Plan for the EOSDIS Core System
193-103-MG3-001	Configuration Management Procedures for the ECS Project
193-105-MG3-001	Data Management Procedures for the ECS Project
193-203-SE1-001	User Interface Requirements Study for the ECS Project
206-CD-001-002	Version 0 Analysis Report
193-208-SE1-001	Methodology for Definition of External Interfaces for the ECS Project
194-401-VE1-002	Verification Plan for the ECS Project
403-CD-001-002	Verification Specification for the ECS Project
409-CD-001-003	Overall System Acceptance Test Plan for the ECS Project
194-415-VE1-002	Acceptance Testing Management Plan for the ECS Project
194-501-PA1-001	Performance Assurance Implementation Plan (PAIP) for the ECS Project
none	Goddard Space Flight Center, Performance Verification Plan (PVP) for the EOS Data and Information System Core System (ECS)

2.3 Information Documents

The following documents, although not referenced herein and/or not directly applicable, do amplify or clarify the information presented in this document. These documents are not binding on the content of the System Integration and Test Plan for the ECS Project Volume 2: Release A:

194-219-SE1-001	Interface Requirements Document Between EOSDIS Core System (ECS) and NSI
194-219-SE1-002	Interface Requirements Document Between EOSDIS Core System (ECS) and the MITI ASTER GDS Project
101-219-SE1-003	Interface Requirements Document Between EOSDIS Core System (ECS) and Landsat 7 System, Working Draft
194-219-SE1-004	Interface Requirements Document Between EOSDIS Core System (ECS) and Version 0 System
194-219-SE1-005	Interface Requirements Document Between EOSDIS Core System (ECS) and Science Computing Facilities
219-CD-006-003	Interface Requirements Document Between the EOSDIS Core System (ECS) and the National Oceanic and Atmospheric Administration (NOAA) Affiliated Data Center (ADC)
193-219-SE1-008	Interface Requirements Document Between EOSDIS Core System (ECS) and Program Support Communications Network, Draft
194-219-SE1-015	Interface Requirements Document Between EOSDIS Core System (ECS) and International partners for Data Interoperability, Preliminary
194-219-SE1-018	Interface Requirements Document Between EOSDIS Core System (ECS) and Tropical Rainfall Measuring Mission (TRMM) Ground System
194-219-SE1-019	Interface Requirements Document Between EOSDIS Core System (ECS) and Earth Observing System (EOS) AM-1 Flight Operations
194-219-SE1-020	Interface Requirements Document Between EOSDIS Core System (ECS) and NASA Institutional Support Systems, 10/94
194-WP-904-002	Multi-Track Development for the ECS Project, White Paper
222-TP-003-005	Release Plan Content Description [for the ECS Project]
540-022	Goddard Space Flight Center, EOS Communications (Ecom) Interface Requirements Document
560-EDOS-0211.001	Goddard Space Flight Center, Interface Requirements Document (IRD) Between the Earth Observing System (EOS), Data and Operations System (EDOS), and the EOS Ground System (EGS) Elements

This page intentionally left blank.

3. System Integration and Test Process

This section defines the process used on the ECS Program to ensure thorough integration and verification of the ECS at the system level as defined by Functional & Performance Requirements (Level 3) and Interface Requirements Documents.

The broad verification approach chosen for System Integration and Test, Build/Thread Testing, is explained. Within this context, the word "Testing" refers to "Test" in the broader context of "verification", e.g., "Test" encompasses the four activities of demonstration, analysis, inspection and test which comprise verification.

The subparagraphs that follow define the System Integration & Test process from three points of view. First, the philosophical approach, Build/Thread (Section 3.1) is described. Second, the processes actually used to plan the testing and how those processes will be carried forward to complete the SI&T effort are discussed (Section 3.2) and third, at a more detailed level, the day-to-day management of System Integration Testing is described in order to highlight the controls applied to the process (Section 3.3). Several special classes of testing required by the uniqueness of the ECS Project are discussed.

The processes described will be performed by the Integration and Test Team (I&TT) within the ECS contractor System Integration and Planning Office (SIP). Supporting this effort are the development segments, the Quality Office and the Configuration and Data Management Department.

Oversight of the System Integration and Test effort is provided by representatives of the Code 505 Integration and Operations office, the IV&V contractor and the ECS Project Independent Acceptance Test Organization.

3.1 System Build/Thread Test Approach

At the system integration and test level within the overall ECS verification process, a build/thread test approach has been chosen. Build/thread methodology relies heavily on the concept of a “thread” - the set of operational procedures, software and hardware that implement a function. Threads are tested individually to facilitate requirements verification and to simplify problem resolution.

The decomposition of release capabilities into threads allows flexible scheduling of development and testing. Such scheduling is influenced by considerations such as:

- Thread Dependencies. For example, basic communication services will be used in most threads. By integrating communications services first, they are available in tested form for all subsequent threads. This also eliminates repetitive use of a communications simulator and reduces test tool costs. By this mechanism, critical core components also get tested in many threads increasing confidence.
- Level of Development Complexity. A thread that requires many custom components may be scheduled late to ensure adequate development time. Threads that contain a significant

amount of COTS will be integrated early while development is in process. This allows more parallelism in the development and integration process.

- Contingency Management. If unforeseen problems arise, threads and builds can often be re-arranged so that overall progress continues. In typical big-bang integration approaches, the entire effort may be stalled.

Threads are allocated to releases. Each thread is intended to be tested in one release, so if only partial capabilities of a function are required in a release the function will be subdivided into separate threads.

Successfully integrated components (software, hardware, and data) that make up a thread are merged with other threads in a gradual buildup of system capabilities - a build. Build tests verify newly available functions in their expanded environment. Regression tests confirm that newly combined functions do not degrade service from previously integrated components.

Groups of builds are, in turn combined to form larger builds until the complete release has been integrated. The final build test consists of end-to-end activities that, while still functionally oriented, approach actual operational scenarios.

3.2 How Testing Is Planned

While Build/Thread is a philosophical approach to testing, this SI&T Plan is a specific application of that approach to the evolving ECS. This section will explain some of that process and chart the path ahead to complete the process for TRMM (Rel A) and subsequent releases. The intent here is to highlight the test-specific activities actually performed rather than to re-document the entire system engineering and verification process.

Initial Build/Thread design began with participation in the formulation of the Release Plan White paper. During this process external mission drivers were examined in order to understand the minimal functionality that had to be present for IR1, TRMM (Rel A) and Release B. This process was iterative in that more information became available with the passage of time. Additionally, initial cuts at functionality provided vehicles for better communication and, hence, understanding of the relationship of mission drivers. Co-incident with the development of the Release Plan a concerted requirements analysis and allocation effort was undertaken. This gave the test team a renewed appreciation for the requirements baseline. Co-incident with this effort the system architecture group began producing white papers explaining the new ECS architecture. These progressed into the individual Logical Object Models (LOMs) which describe the allocation of services at the logical level.

3.2.1 Thread/Build Analysis

Using the information base thus gathered, suggestions for threads were formulated on the basis of the functions defined in Table 10 of the Release Plan. These were examined in the light of the criteria discussed in Section 3.1. Based on this, initial aggregations of the threads into builds were made and discussions with representatives from the development organizations to gather their assessment of reasonableness and correctness.

After several iterations, preparation of the written descriptions of the threads and builds was begun. The act of concisely describing the thread and the verification of it ends up being a

fundamental part of the engineering process. In some cases, attempting this proved that the thread content was poorly chosen in that there may be too many unsatisfied external dependencies or that the results of the threads actions aren't sufficiently externalized to allow verifications. Such discoveries, caused revision to the build/thread plan resulting in the version of the plan submitted.

Each thread and build in the document is taken to the final level of detail by enhancing the description of the test flows to take into account the details of the implementation of the design. If the size of the thread warrants, multiple test *sequences* within a thread may be created. Sequences serve to enhance management of the testing activity by dividing the effort into smaller tasks that can be prepared and executed independently.

3.2.2 Test Plans

Given the increasing level of detail in the design of the release, better knowledge of the interfaces between components is available. This allows individual threads to be assessed for their ability to standalone during testing. If because of design progress, a thread is found to have unexpected dependencies that cannot be met, two options exist. First, restructuring of a portion of the build/thread plan can occur. This can often be done with only minor impact on the overall effort. The second option is to provide test tools (stubs or drivers) to meet the needs of the thread. Early in the integration process, such stubs or drivers are more numerous because many of the threads are dedicated to providing mainly infrastructure.

Since a thread, or a sequence within a thread, defines a set of actions to be performed to evoke a desired response, the next step in the detailed planning process is to determine the range of test values to be used to exercise the function. A set of inputs is chosen to assess functionality. This is supplemented by values at the edge of the acceptable range and values beyond the acceptable range. For each set of input conditions, an expected set of outputs is required. The pairing of a set of inputs and outputs to be applied to a thread or sequence are considered to be a *test case.* Sufficient test cases are formulated to achieve a certainty of correctness commensurate with the criticality of the function under test.

When this is accomplished for all builds and threads, this plan is submitted for PDR/IDR. With this, the end of the test planning phase occurs and the generation of test procedures (step by step instructions) begins.

3.2.3 Test Procedures

The next stage in the test process is the generation of detailed test procedures (DID 414/VE1) based on the plan. The PDR version of the plan essentially forms an outline for the test procedures. The test procedures are a step-by-step set of instructions for the actual execution of the testing required to integrate and verify the thread. Additionally, the procedure has a concise definition of the test environment for the activity. Like the test plan, the procedures are developed in stages as more and more information becomes available during the development life cycle. Before code first becomes available after CDR, detailed design information and initial user manuals are used to begin the procedures. Another key activity during the period is the development of test tools and the collection/production of test data.

As software becomes available, initial procedures are checked out by executing them against the system software. Such execution attempts to exercise the system and the procedures themselves. Problems that occur can be the result of either errors in the procedures or errors in the product. The test team members work with developers to investigate problems and find a resolution. This cooperative process continues until the discrepancy rate moderates and until there is confidence that the procedures are correct. This generally occurs before the system under test is complete and performing correctly. The test procedures are delivered at this point in the progress.

3.2.4 Test Conduct

After procedure delivery, test conduct begins. Test conduct is the execution of the test procedures against a software baseline that is under configuration control. The goal of test conduct is the verification of requirements through successful execution of the test. During this time, portions of test procedures may be executed out of sequence to concentrate on particular parts of the system. Test conduct continues until all parts of all test procedures have been successfully executed.

Test conduct culminates with the formal execution of the test procedures as a whole before appropriate witnesses. Required witnesses include representatives of the Quality Office, ECS Project Management and ESDIS Integration Office. As always, the authority to witness may be delegated or waived on a case-by-case basis.

3.2.5 Consent-To-Ship Review

At the completion of the formal execution of the System Integration Test, a Consent to Ship Review (CSR) is held. As with all test reviews the CSR serves two purposes. First, it determines if the current activity, SI&T, has completed successfully. Second, it assesses whether the next activity, Acceptance Test, may begin. The CSR is a formal meeting chaired by the Test Lead at which the following are presented:

- Initial Test Results - Based on a quick-look analysis of test data the outcome of the testing is presented.
- Deviations from Test Procedures - If during the demonstration, any deviations from the printed procedures were necessary, they are explained and discussed to establish that they did not invalidate the test execution.
- Non-conformance Report (NCR) Status - The status of all open NCRs is presented along with recommendations for their disposition.
- Configuration Management Status - The CM organization reports on the status of the product baseline.
- Recommendation - Based on the previously presented material the Test Lead recommends the acceptance/rejection of the test.

3.3 How Testing Is Controlled

This section discusses the basic, day-to-day management of the verification process. The fundamental steps to be described here are performed during test preparation, conduct and the formal demonstration. As the test progresses and matures the degree of formality and the

frequency of process checks increases to insure timely completion and thorough testing. For the sake of this presentation, the activities of a theoretical "day' of testing is described. It should be noted that this "day" is not necessarily an actual calendar day. During the preparation phase the unit of management might be a week, while during the formal demonstration it might actually be a day or even part of day.

A testing session begins with a pre-test meeting at which the planned activities are discussed, assignments are made and the configuration for the test environment specified. The readiness of the environment to support the activity is assessed. Other than during critical times and during the formal demonstration, the pre-test meeting is held as the last part of the previous sessions post-test meeting.

The key to effective testing is the documentation or logging of what is done. Good note taking serves to:

- Capture the state of the product under test (e.g. version numbers, concurrent activities).
- Capture the chronology of actions - without this, reported problems cannot be recreated and understood to allow correction.
- Provide an audit trail - this allows verification of coverage and collection/analysis of metrics.

At a minimum this information will be captured in individual test logs. Correlation of widely dispersed events is going to be especially difficult because the ECS is a widely distributed system. Development of on-line tools are being considered to supplement hard copy notebooks and to allow easier time correlation and collaborative testing. Extensions to public domain products like NCSA Collage may be feasible.

Control of the test environment is crucial to the test process. Included are software, hardware and "user/operational" environments. Software and hardware states are tracked and controlled by the CM system. During testing, the Test Lead is delegated authority from the Configuration Control Board to manage the test environment. The process used is discussed in the paragraphs on Non-conformance Reporting and Corrective Action (NCRCA). An area of difficulty in the distributed environment is the control of the user/operational environment. During development and the early phases of test preparation, test and development will share the same hardware and system software environments. Aside from the impact on performance due to resource contention, updates to shared data, etc. can occur. This will be handled through close coordination with development. During the later stages of testing, the environment will be controlled by excluding other than test users, through strict access controls managed by the test lead.

Whenever problems occur they are entered into the ECS NCRCA system. All problems are entered into the NCRCA, assessed and dispositioned. While invariably duplicate and erroneous problems are entered, mechanisms are in place to quickly and efficiently detect them and close them out. This approach tends to insure that no problems are overlooked. The NCRCA system also supports the collection of metrics.

System Integration Testing maintains its own list of Non-conformance Reports (NCRs) within the NCRCA covering problems that occur during the time the product under test is controlled by SI&T. Thus, if, during preparation, a test team member discovers a problem, it is entered into the NCRCA as a System I&T problem. If during assessment (discussed later), it is determined to be

caused by a component that has not yet been turned over to SI&T, the problem is transferred to the Segment Integration and Test portion of the NCRCA for resolution.

At the end of each test "day", a post test meeting is held by the test lead. Attending are test team members, development support personnel and Quality Office representatives. The purpose of the meeting is to assess progress, plan future activities and review the status of NCRs. This review includes:

- New NCRs - Each new NCR is presented (usually by the author). The problem encountered is described along with the author's recommendation for priority. The test lead assesses the NCR and assigns a disposition. Most often at this stage in the process the problem is assigned to a development representative for further investigation. The test lead assigns a priority.
- Open NCR Status - Status is sought on open NCRs. The developer support returns information of the problems that are being fixed in support of the testing. When a fix is completed, it is reported and the status of the NCR updated to "Resolved". Before the NCR may be closed, two important things must happen. First, the Test Lead must allow the fix into the test baseline via the CM system. On a case-by-case basis, proposed fixes are considered for priority and impact on the test as a whole to determine the strategy for their inclusion. Some fixes are held until a related group is ready, others may be incorporated individually. Second, once the fix has been installed into the test baseline a test team member must verify the fix (usually by re-executing the procedures (steps) described in the NCR)). If this testing is successful, the NCR is closed at the discretion of the Test Lead.

3.4 Roles and Responsibilities

This section describes the responsibilities of different organizations in the integration and verification of the ECS.

3.4.1 System Integration and Test

The Systems Integration and Test (SI&T) organization is part of the System Integration and Planning (SI&P) Office which additionally has responsibility for Engineering Planning, Technical Assessment and Standards, V0 and External Activities, Interface Engineering, Co-ordination and Control, and Performance and Cost Modeling.

Figure 3-1 shows the ECS contractor's System Integration and Planning (SI&P) Office and System Integration and Test organization (SI&T).

The Integration and Test Team (I&TT) is primarily responsible for the ECS system integration and verification. Upon successful completion of testing, the I&TT will hold Consent to Ship Review (CSR) for delivering the software to the Independent Acceptance Organization Test (IATO) via the CM organization.

The Independent Acceptance Test Organization (IATO) is primarily responsible for the systems acceptance testing, which occurs subsequent to the system integration and test activities. Implied in this responsibility is the installation of the releases at the remote sites. Support for the Independent Verification & Validation (IV&V) contractor is managed via the IATO.

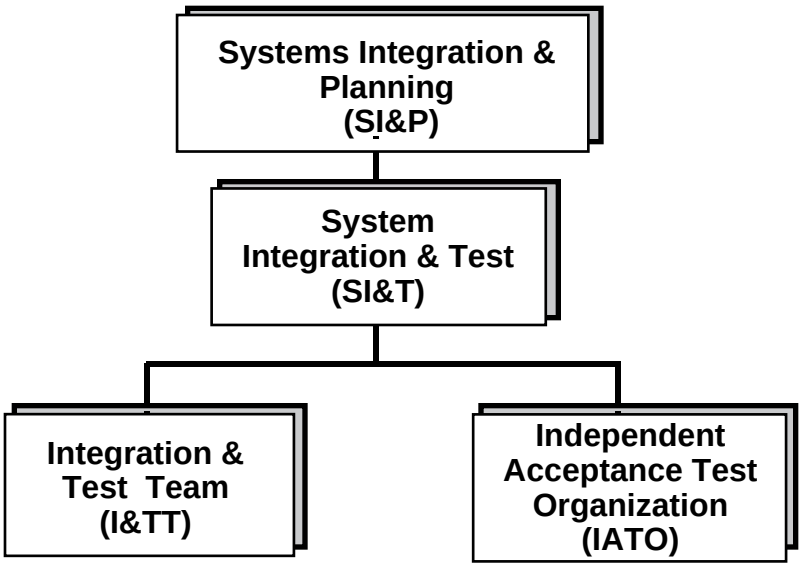

Figure 3-1. Organizational Chart

Table 3-1. Configuration Controlled Software Baselines for Test Program

Test Level	Performed By	Documented In	Code Location	Completion Event
Unit	Developer	Unit Dev. Folder	Developer's Files	Test Readiness Review
Segment I&T	Segment Tester	S/E Int. & Test Plan (DID 319)	Segment CM Library	Element Test Review
System I&T	SI&P - SI&T	Sys. Int. & Test Plan (DID 402)	System CM Library	Consent to Ship Review
Acceptance Test	IATO	Acceptance Test Plan (DID 409)	IATO Library at Sites	Release Readiness Rev.
System	IV&V	TBD	O&M Library at Sites	TBD

3.4.2 Configuration Management

All deliverable application software and software test tools will be controlled via the ECS Configuration Management. tool (ClearCase) which is administered and controlled by the Configuration Management Organization. During the entire development process the CM organization provides the software library function. As the SW product moves from development, to Segment I&T, to System I&T and ultimately to Acceptance testing, it is within the CM tool. The control authority over the baseline is delegated to the appropriate individuals during the process. In the case of System I&T, the Test Lead has authority to control changes to the CM baseline under test. Test Procedures and data used during the test will also be baselined and controlled.

3.4.3 Quality Assurance

A Non-conformance Reporting and Corrective Action (NCRCA) system managed by the Quality Assurance Office is used to control discrepancies identified in both documentation and software. A Non-Conformance Report (NCR), is used for any departure from design, performance, testing or handling requirements that affects hardware, software or documentation. The process for Non-conformance Reporting was described in Section 3.3.

The Quality Office oversees many facets of the testing process through inspections of work in progress. At formal demonstrations the Quality Office witnesses the test activities to insure compliance with written procedures. Quality Office responsibilities are described in the Performance Assurance Implementation Plan (PAIP), DID 501/PA1.

3.5 System I&T in a Multi-Track Environment

For a specific release, two main development processes will be used: the Formal Development Process and the Incremental Development Process described in the Multi-Track Development White paper. Figure 3-2 from that document shows the relationship between the two processes.

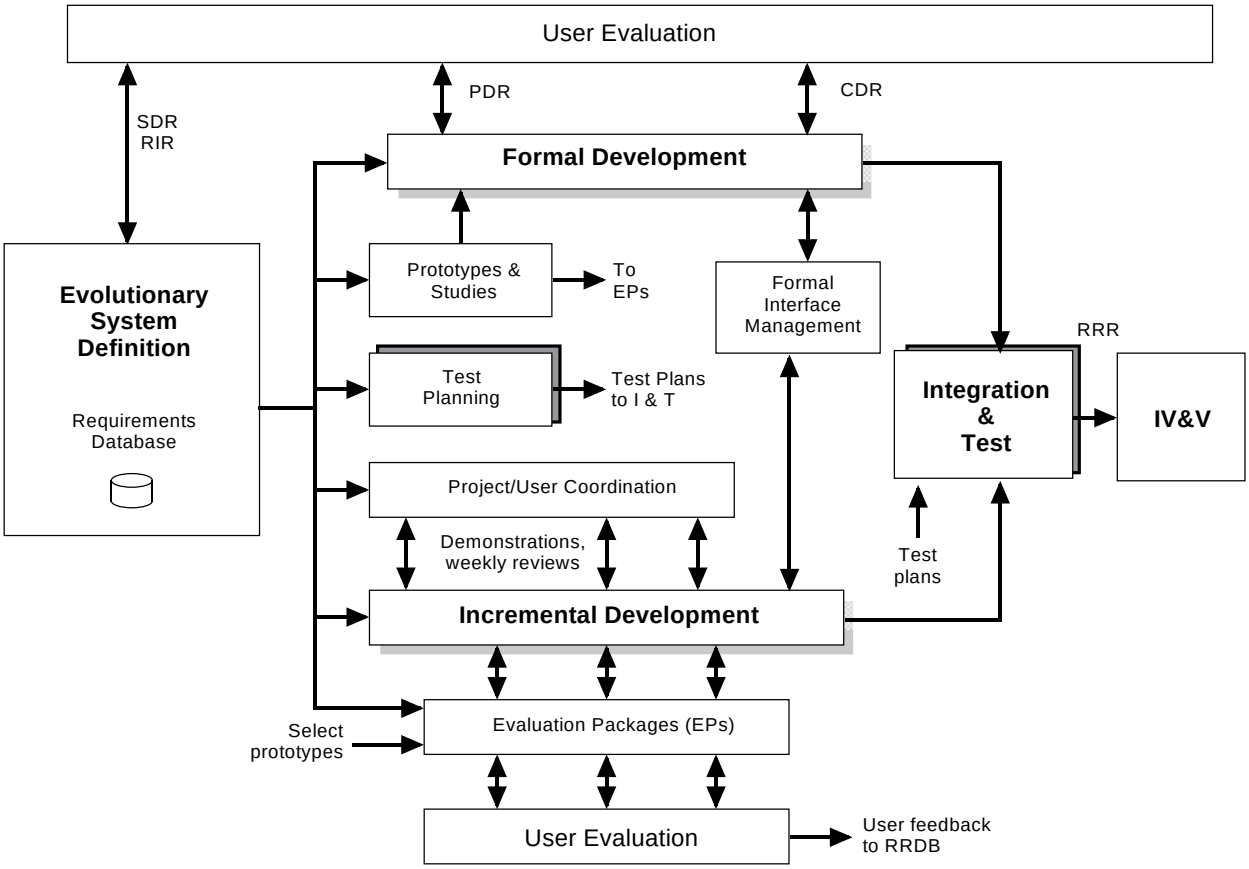

Figure 3-2. System I&T and Multi-Track Environment

Both formal and incremental development tracks will be implemented to: 1) assure compliance with requirements, 2) provide traceability of requirements allocation to tracks, 3) implement an

integration process that brings the separately developed pieces together into an integrated whole, and 4) provide a process for control of interfaces that supports integration.

From a System Integration & Test point of view, the Incremental Path can be thought of as nothing more than another development methodology. Software from this path enters the test process at the Test Readiness Review (TRR) for integration at the segment level. This means that it has rejoined the normal development flow and should be indistinguishable from formally developed software.

In practice, software from the incremental path is different. First, because its requirements analysis and objectives determination process is handled out of the formal path and SI&T planning for the increments is less precise early in the formal process. The Integration team participates with the segment test organizations in the planning and testing of the Evaluation Packages but do not control the test environment. This means that the I&T personnel have earlier contact with and influence over incrementally developed products than with formally developed ones. This knowledge and involvement means that we have the ability to react quickly to evolutionary changes during the incremental process. Any such changes will be reflected as updates to formal track test documentation, as necessary.

3.6 Integration and Interface Testing

The integration of elements and segments and the integration of the ECS with external systems is a fundamental part of the System I&T process. The interface requirements for the ECS will be documented in various interface requirements and control documents, both internal and external:

- Methodology for Definition of External Interfaces (DID 208/SE1)
- Interface Requirements Documents (DID 219/SE1)
- External Interface Control Documents (ICDs) (DID 209/SE1)
- ECS Internal ICDs (DID 313/DV3)

Interfaces are exercised and tested as a part of the build/thread testing process. The build-up of the system described in this document is fundamentally one of aggregation of components (threads and builds).

To perform the integration and interface verification, simulators and simulated data flows will be used until actual system capabilities exist on both sides of the interface. Version 0 data sets will be valuable as a set of early test data. For external interfaces it may be necessary to remain with simulators throughout the test process because of schedule (and scheduling dependencies). System Integration and Test is committed to performing engineering-level early interface testing with external systems whenever possible. Such activities will be arranged through the interface definition process and the Ground System Integration Working Group.

3.7 Test Tools

Four areas of applicability of test tools to the ECS Program have been identified. First, tools are needed to support automated test planning and management. These tools assist in the development and tracking of test cases, test data, test results, as well as the mapping of requirements to test cases. The Requirements and Traceability Management (RTM) tool has been

selected by the ECS project to support the requirements management process. The RTM tool provides the means to record all relationships and dependencies between requirements, documentation, releases, services, and more specifically, test specifications. RTM assists systems engineers in defining requirements, assigning them to release, and mapping them to formal test cases. In addition, test results will be recorded and mapped to requirements within RTM, so that at any point, the status of the test and verification of a specific requirement can be checked.

Second, tools are needed to simulate immature interfaces, especially for systems external to ECS. External interfaces refer to systems outside of the scope of the ECS contract. The interfaces may be to systems already in existence, systems that are being built as part of the overall ESDIS project, or systems being built by other Government agencies or other countries. Simulators for external interfaces generate and transmit data streams in the identical format that represent the specifics of the real system's data stream.

Third, tools are needed to automatically execute test procedures or scenarios. Included in this are Remote Terminal Emulators (RTEs), to emulate live users, data generators, to generate simulated input data, and programmable test languages. For RTEs, a tool is needed that is capable of emulating the maximum number of users that ECS is required to support at one time. For data generators, simulated data sets (in Level 0 format) from each of the instruments and possibly from existing satellites will be needed. A programmable test language having some simple command structures that is capable of controlling interface simulators as well as the RTE is needed.

Finally, data reduction and analysis tools will be needed to process and summarize the large amount of test data anticipated by the ECS Program. Data reduction and analysis tools are utilities designed to analyze test output data, including utilities to compare test output to benchmark data. A sophisticated file compare utility is needed to compare expected test results to actual test results. A data reduction utility is needed to reduce large amounts of output data to some meaningful evaluation of the data's quality.

The testing environment will evolve as the ECS is developed and the Program matures. For system integration and test, the bulk of the testing will be performed within the ECS Development Facility (EDF), located within the Hughes complex. Many simulated data streams and simulated functional capabilities will be used due to the evolving maturity of the ECS.

Since most of the tests at the EDF will have a large simulated component, it will be important to accurately record and be able to accurately reproduce test conditions, test data streams, test workloads, etc.. Test tools assist in managing and controlling the test environment to make reproducible and controllable test conditions possible.

3.8 Tool Kit Testing

Tool kits provide a controlled interface into the services provided by the ECS. These interfaces are provided at four levels: User, Software, Algorithm and Application Interface. Because these are fundamentally different, the testing done on each will be significantly different. These are discussed separately in the paragraphs that follow.

3.8.1 User Interface Tool Kit Testing

User Interface Tool Kits provide the functionality that a remote user must employ to access the user oriented services offered by the ECS. With this software the remote user has access to the full graphical interface allowing data search, data browse and data access. Other, simpler interfaces accomplished through remote login are also provided. The Flight Operation Segment (FOS) provides the Instrument Support Terminal (IST) that is envisioned to allow the field investigator limited interaction with, and control of, individual instruments on the ECS platforms. The IST is scaled to allow operation on modest workstations that would likely be available at more remote sites.

Testing of the user interface tool kits is largely indistinguishable from the normal mainline verification of the Release as a whole. Tool kit software is identical to that which will run in the local terminals at the EOC and the DAACs themselves. The only differences requiring specialized testing are the potential impact of the network connection and the degree of equipment compatibility. The ability of the tool kits to mitigate the impact of the network is verified by using a remote login through the network while testing is on-going in the EDF. Through careful attention to message routing, messages can be directed out through long loops on the Internet even though the connection could be made directly on the local LAN.

3.8.2 Algorithm Interface (PGS) Tool Kit Testing

The PGS Tool Kits exist in two forms. The first is delivered to scientists at the Science Computing Facilities (SCFs) to provide an environment in which algorithms destined for operational use within the ECS Product Generation System (PGS) will be developed and tested. The second form of the PGS Tool Kit replaces the first when the SCF-developed algorithm actually executes in the PGS at a DAAC. These two cases are illustrated in Figure 3-3. Both tool kits provide identical calling sequences to the science algorithm.

The SCF version of the PGS Tool Kit as illustrated in Figure 3-4, is delivered before formal delivery of the releases to allow algorithm developers an early start. Thus, the tool kit is tested separately through use of a benchmark algorithm and test drivers.

Since there is no integration of the tool kit with other parts of the ECS, this testing is not accomplished as a part of System Integration Testing and is not described in this plan. In fact, SI&T cooperates with Segment I&T to perform SCF toolkit testing. The final release of the Toolkits will be acceptance tested by the IATO.

The operational version of the PGS Tool Kit as illustrated in Figure 3-5, is delivered as part of a complete release and is tested with that release. The tool kit is tested coincident with the testing of the PGS through the use of one or more benchmark algorithms.

The focus of this testing is two fold. First, it must prove that the interfaces to the algorithm meet the interface definitions of the PGS Tool Kit Specification. Second, it must prove that the PGS interfaces with the algorithm for the delivery/receipt of data and the scheduling/control of algorithms.

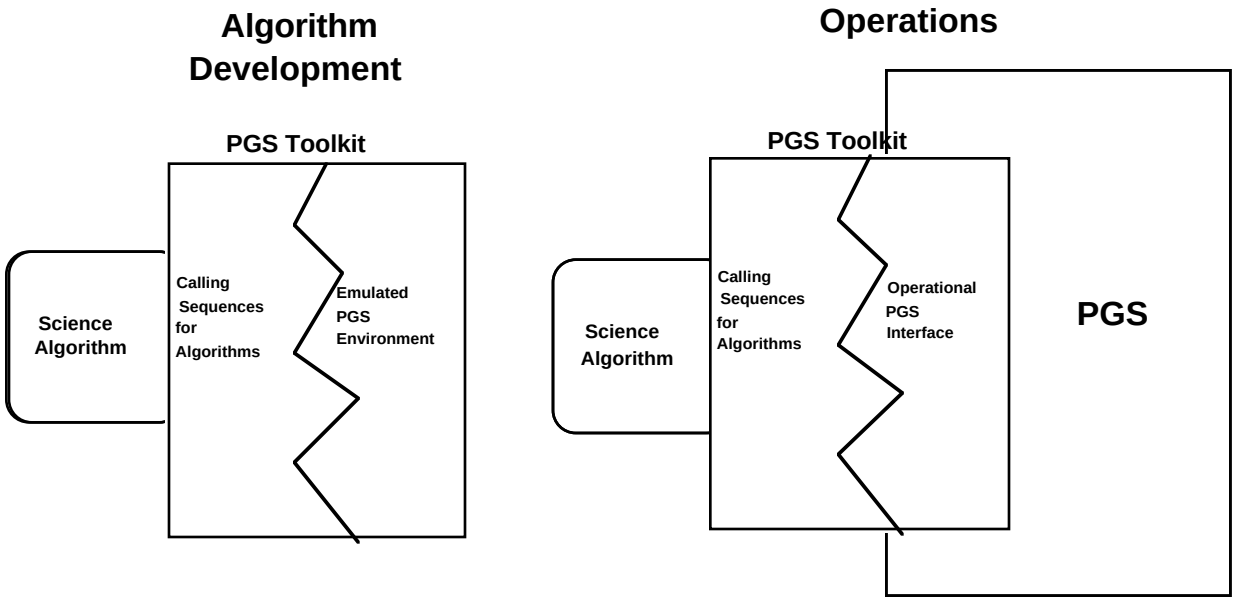

Figure 3-3. PGS Tool Kit Environments

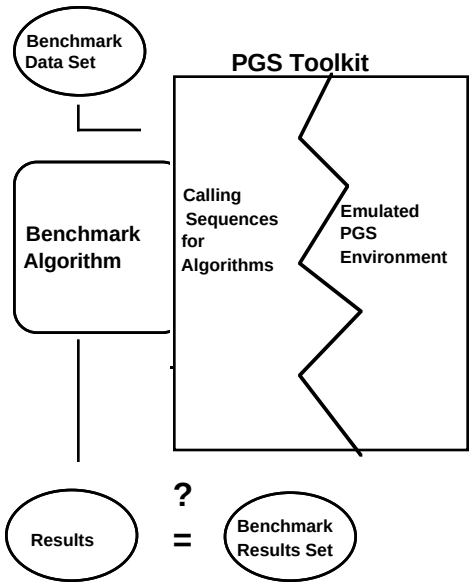

Figure 3-4. PGS SCF Tool Kit Test Environment

3.8.3 Applications Program Interface (API) Testing

Applications Program Interfaces (APIs) are provided by the ECS to allow properly authorized, externally generated software systems access to ECS Segment/Element functions at the client/server level.

Testing of these APIs will involve the use of simple driver programs that validate compliance to the interface specifications. The operations attempted through the APIs will be chosen to explore the full range of the interface's parameters with emphasis on the most probable usage. The other necessary testing emphasis must be on verifying that the API protects the core ECS system from transgressions by the user processes.

3.9 Schedules and Dependencies

Documentation vehicles, such as the ECS Intermediate Logic Network, that are regularly released will be used as the vehicle to disseminate current schedules.

Your attention is called to the Build/Thread diagram in Section 5. While it contains no temporal notation, it implies a scheduling relationship among their component parts. As the release cycle progresses, they will be annotated to represent meaningful schedules.

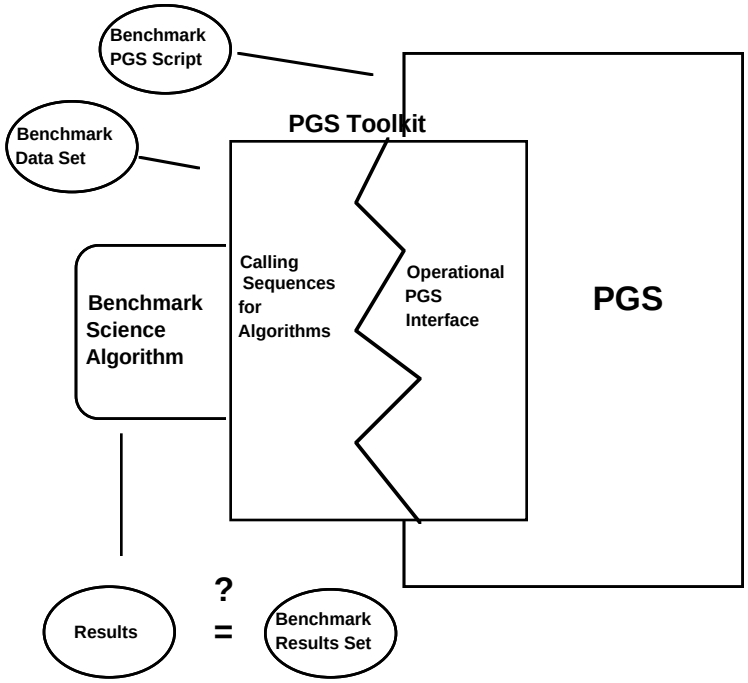

Figure 3-5. PGS Testing Environment

4. IR-1 System Thread and Build Plan

The IR-1 System Thread and Build Plan can be found in the corresponding 402 document titled 'System Integration and Test Plan for the ECS Project Volume 1: Interim Release 1 (IR-1)

This page intentionally left blank.

5. TRMM (Rel A) System Thread and Build Plan

The system build/thread plan created to support the TRMM release was formulated using information in the Release Plan Content Description (222-TP-003-005 - FB9403V5).

The TRMM release will support end-to-end ECS system operations.

TRMM will support ingesting and verification (data checking) of Level 0-4 data for the TRMM platform, ingesting and pre-processing of ancillary data, ingesting of initial V0 HDF data, and ingesting of other native formats, and the capability to send/receive data (to include documents, algorithms, binary data) using various network protocols (TCP/IP, ftp, RPC).

TRMM will support a local heterogeneous data server environment (data server management) at each DAAC activated for the TRMM release (MSFC, LaRC, GSFC) and the EDC. Data server management encompasses maintaining top level information (guide, directory, inventory data) of the data written to the server(s) (archive), the permanent storage of all ingested data/products, and the permanent storage of all higher level products produced during system processing. Advertising of all data existing within each DAAC's data server (archive) environment to the Local Information Manager (LIM) level is provided. As data is stored to or deleted from the archive, the LIM data is updated to reflect these changes.

Data processing to support the generation of metadata and higher level products is provided. Production processing includes the ability to generate a site specific processing plan (schedule) and the ability to modify/update the plan to support user/client requests processing requests on a priority basis.

Each segment (FOS, CSMS, and SDPS) is responsible for providing the software to support the TRMM release.

FOS is responsible for providing the software to support early interface testing of the AM-1 spacecraft. This software includes the capability to plan and schedule both spacecraft and ground resource activities in support of spacecraft operations and real-time data receipt. The capability to generate spacecraft command and instrument loads, uplink of the command and instrument loads, real-time receipt and monitoring of spacecraft engineering and health and safety data, and real-time receipt of science instrument data is supported in the TRMM release.

CSMS is responsible for providing the software to support basic system and network communications support (both locally and remotely), network and system security, system monitoring (to include the capability to locate, isolate, correct fault conditions), configuration management, and history logging/reporting of system activities.

SDPS is responsible for providing the software to support end-to-end system processing and distribution of data. This includes the capability to ingest data, write it to a staging area, validate the data, perform pre-processing (if required), perform reformatting or data conversion (if required), process the data and archive the data. Advertising of data written to the archive to the LIM level is supported in the TRMM release. A methodology for users/clients to generate

queries, distribution requests, and perform searches as well as the ability to provide interim status updates to users/clients for pending user/client requests is required

Each segment is responsible for providing any tools required to support data comparison, data manipulation, and statistical analysis.

System Integration and test will verify the functionality/capability of the software developed by each segment using a build/thread philosophy. Elements will be tested independently as a thread with stubs being provided for interfaces/functionality not yet provided. Testing will demonstrate that each element turned over to system integration and test meets Level 3 requirements specifications. Completed threads will be integrated into a "build" test at the system level. Integration testing of a build will verify that all components function properly when integrated together, that no functionality has been lost as a result of the integration, and test new capabilities as a result of the integration effort. Interface testing will verify the handshaking between each element.

The full up system test will verify the ability to perform concurrent system activities to include: user/client logon, registration, data searches, queries, distribution requests, status requests, ingesting of data, on-going system processing, recovery from fault conditions under normal system operations, and implementation of SMC directives and schedule adjudications.

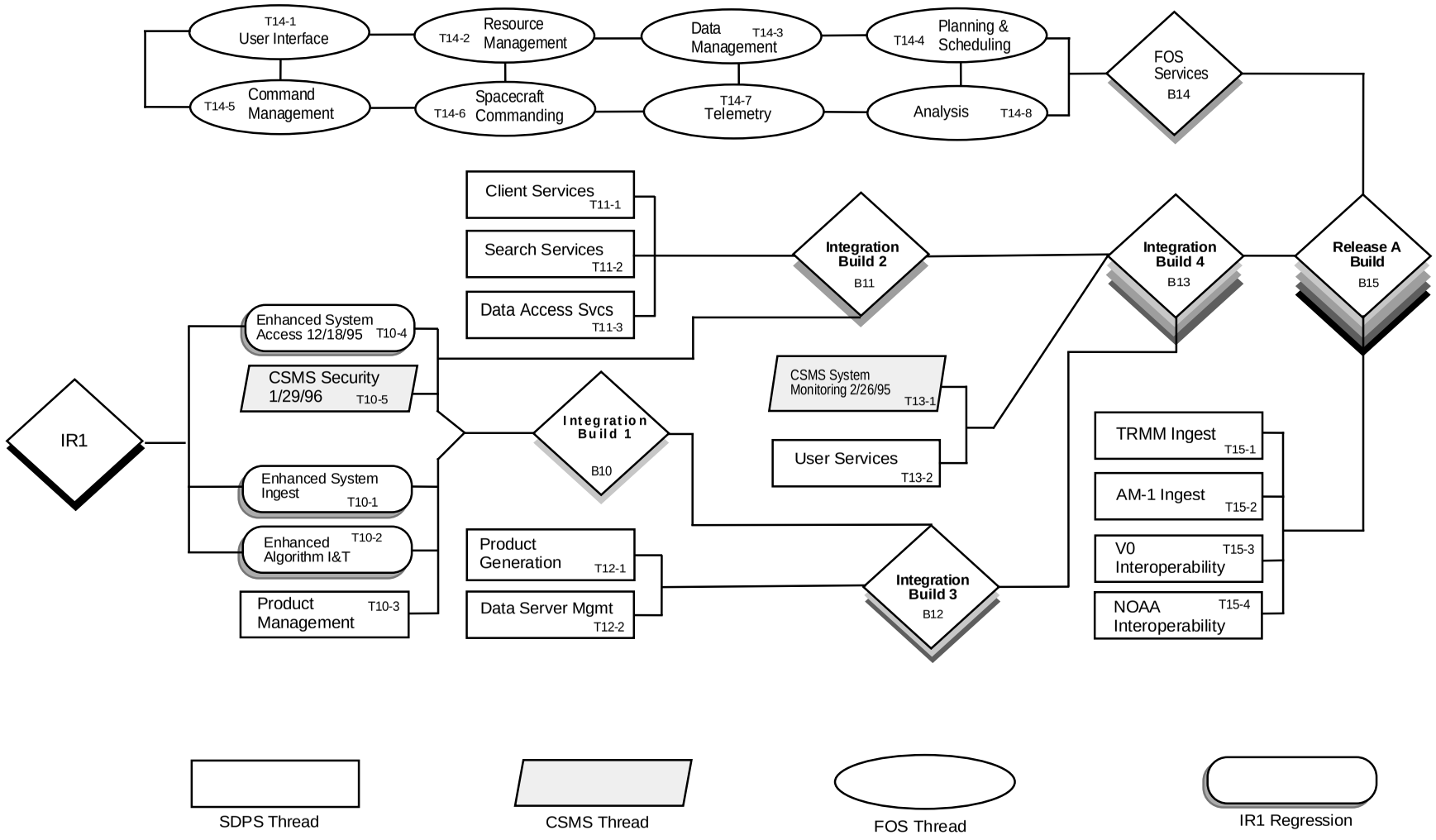

Figure 5-1. TRMM (Release A) System Build/Thread Plan

12/13/94

5.1 TRMM (Rel A) System Threads/Builds

5.1.1 Enhanced System Ingest Thread (T10-01)

5.1.1.1 Thread Objectives

The objectives of the enhanced system ingest thread are:

- Provide the IR1 Basic Ingest functionality.
- Provide the ability to ingest ancillary data.

  Provide the ability to preprocess ancillary data into a usable form (this will result in generation of calibrated data products).
- Provide the V0 ingest capabilities.

5.1.1.2 Thread Test Description

Regression testing of the basic ingest functionality provided in IR1 is performed. Metadata and ancillary data are ingested into the staging area of the ECS system.

Testing verifies the capability to ingest data, verify the data, and log the ingest transaction. Metadata is extracted and logged and then the data is archived into the ECS system. Once this has been completed, the ingest process logs the completion status in the ingest log and issues a completion notice to advertise data availability.

The capability to ingest V0 data into the ECS system is verified. Testing also verifies the capability to generate calibrated data products.

Dependencies: (If Applicable)

In order to verify the objectives of the enhanced ingest thread, the following interface and functional capabilities are required:

- Interface to the data server to archive ingested data.
- Ability to perform pre-processing of ancillary data to produce calibrated data products.
- Connectivity to GSFC, LaRC, and MSFC V0 networks.

Test Support Requirements

- Hardware:
  - workstation
  - operational system
  - physical media (4 & 8 mm tape, 3480/3490 tape, CD ROM, and at least one document for hard media ingest)
- Software:
  - operational software
  - monitoring tools for data ingest

- command procedures consisting of multiple ingest requests
- analysis tools or utilities for formatted and compressed data

• Data:

- The following data is needed to perform ingest thread tests:

  L0 science data

  V0 data

  Correlative data (used for calibration, validation and verification of EOSDIS science products)

  ancillary data

5.1.1.2.1 Sequence 1 - Regression Test of IR-1 Ingest

The following series of tests verify the ability to indicate its readiness to accept data, IR-1 ingest data, validate data integrity, write the data to the staging area, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 IR-1 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of IR-1 data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: IR-1 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: IR-1 data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 Identifying the Incoming data set

This test case verifies the ability to process the header information to identify the incoming data set.

Input: Data sets of various data types.

Output: Ingest data placed in data staging area. Ingest log contains an entry for the ingest of data and staging of the data.

Expected Results: The data set is identified (i.e., originator, data type, data class) and properly logged.

5.1.1.2.2 Sequence 2 - Ingest Error Checking

The following tests verify that error log entries and error status messages are properly logged for all errors encountered during the ingest process. Issuance of retransmission requests for bad or missing data is verified. Possible errors that can be encountered during ingest are: ingest request with header errors, ingest request with null values, ingest request with invalid formats, ingest request with incorrect data, and ingest request without authorization.

Test Case 1 Valid Header Information

This test verifies that when data is received with valid header information, the data is ingested in the data format agreed upon in the data availability schedule.

Input: Ingest data with valid header information. Ingest monitoring interface tool.

Output: Header information validated, data ingested, ingest log contains an entry for the ingest initiation, header verification, and successful ingest of valid data.

Expected Results: Header information validated, data successfully ingested.

Test Case 2 Invalid Header Information

This test verifies that when data is received with invalid header information, the data is not ingested and the recipient issues a retransmission request to the originator defining the error(s) encountered during the ingest process.

Input: Ingest data with invalid header information. Ingest monitoring interface tool.

Output: Header information determined to be invalid, retransmission request generated, ingest log contains an entry for the ingest initiation, header validation failure, and the generation of a retransmission request to the originator.

Expected Results: Header information determined to be invalid. Retransmission request generated.

Test Case 3 Verify Mandatory Data Fields

This test verifies that required data fields do not contain "null" values.

Input: Ingest data with "null" values in required fields. Ingest monitoring interface tool.

Output: Data validation fails due to missing required fields of data, ingest log contains an entry for the ingest initiation, data validation failure, and the generation of a retransmission request to the originator.

Expected Results: Data content determined to be invalid. Retransmission request generated.

5.1.1.2.3 Sequence 3 - Ingest Converting/Reformatting and Compression of Data into Approved ECS Format

The following series of tests verify the ability to successfully convert/reformat data into an approved ECS format and to compress the data prior to storage.

Test Case 1 — Converting data into ECS format

This test case verifies the ability to convert ancillary data to an approved ECS format prior to data storage.

Input: Ancillary data requiring conversion to ECS format. Data conversion tools.

Output: Ingest log contains an entry for the ingest initiation, successful ingest of the ancillary data, and the conversion of the data into an approved ECS format.

Expected Results: Ancillary data converted to approved to ECS format.

Test Case 2 — Data compression

This test case verifies the ability to compress data prior to storage.

Input: Reprocessed ancillary data. Data compression tools.

Output: Ingest log contains an entry for the compression of the ancillary data.

Expected Results: Ancillary data compressed and ready to archive.

5.1.1.2.4 Sequence 4 - Metadata Validation and Transaction Status

The following series of tests verify the ability to ingest metadata in an approved ECS format (metadata shall be checked at a minimum for: presence of required fields; error-free input; correctness of the data set granule size; and other checks as required) and provide a status of the transaction.

Test Case 1 — Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received and to provide a status of the transaction.

Input: Metadata resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 2 Invalid Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received.

Input: Metadata resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 3 Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated IR-1 product. Originator is notified of the re-generation of metadata for the associated product data.

Input: Metadata resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

5.1.1.2.5 Sequence 5 - Ingesting Ancillary/Calibration Data

The following series of tests verify the ability to ingest ancillary data, verify the contents, and perform pre-processing of the ancillary data to produce calibrated data products.

Test Case 1 Valid Ancillary Data

This test verifies that valid ancillary data can be ingested, pre-processed, and archived accordingly.

Input: Valid ancillary data, ingest monitoring tool, pre-processing tool.

Output: Ingest log contains an entry for the ingest initiation, validation of the ancillary data set, initiation of the pre-processing of the ancillary data, completion of pre-processing, and generation of the completion notice.

Expected Results: Ancillary data validated, pre-processing performed and originator notified of data availability.

Test Case 2 Invalid Ancillary Data

This test verifies the capability to recognize invalid ancillary data and generate a retransmission request to the originator.

Input: Invalid ancillary data, ingest monitoring tool.

Output: Ingest log contains an entry for the ingest initiation, invalidation of the ancillary data set, inability to perform pre-processing, and generation of the retransmission.

Expected Results: Ancillary data determined to be invalid, pre-processing never initiated, and retransmission request generated.

Test Case 3 Archiving Calibration Data Set

This test verifies the ability to ingest calibration data sets and perform permanent archiving.

Input: Calibration data set to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of calibration data set ingest, call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Calibration data ingested, written to the staging area, written to permanent archive, and the staging disk space de-allocated.

5.1.1.2.6 Sequence 6 - Ingesting V0 Data

The following series of tests verify the ability to ingest V0 data from each V0 network host supported in Release A.

Test Case 1 GSFC V0 Access

This test verifies the capability to ingest V0 data from the GSFC V0 network.

Input: V0 data from GSFC. Ingest monitoring tool.

Output: Ingest log contains an entry for the ingest initiation, validation of the V0 data, and generation of the completion notice.

Expected Results: Ingest of GSFC V0 data successful.

Test Case 2 LaRC V0 Access

This test verifies the capability to ingest V0 data from the LaRC V0 network.

Input: V0 data from LaRC. Ingest monitoring tool.

Output: Ingest log contains an entry for the ingest initiation, validation of the V0 data, and generation of the completion notice.

Expected Results: Ingest of LaRC V0 data successful.

Test Case 3 MSFC V0 Access

This test verifies the capability to ingest V0 data from the MSFC V0 network.

Input: V0 data from MSFC. Ingest monitoring tool.

Output: Ingest log contains an entry for the ingest initiation, validation of the V0 data, and generation of the completion notice.

Expected Results: Ingest of MSFC V0 data successful.

5.1.1.2.7 Sequence 7 - Ingesting Algorithms

The following series of tests verify the ability to ingest algorithms, validate header content/descriptors, write the data to permanent archive, and generate an ingest completion notice to the originator of the ingest.

Test Case 1 Algorithm Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of the algorithm, initiate data ingestation, and verify header content/descriptor.

Input: Algorithm to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of algorithm ingest and the validation of the header content/descriptor information. Data contained in the temporary staging area.

Expected Results: Algorithm data ingested, header content/descriptor information validated, and written to the staging area.

5.1.1.2.8 Sequence 8 - Ingesting Non-EOS Ancillary Data

The following series of tests verify the ability to indicate ingest's readiness to accept data, ingest of Non-EOS ancillary data, validation of data integrity, ability to perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 Non-EOS Ancillary Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of Non-EOS ancillary data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: Non-EOS ancillary data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: Non-EOS ancillary data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 Non-EOS Ancillary Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of Non-EOS ancillary data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid Non-EOS ancillary data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: Non-EOS ancillary data ingest terminated and originator receives retransmission notification.

Test Case 3 Non-EOS Ancillary Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with Non-EOS ancillary data and to provide a status of the transaction.

Input: Non-EOS ancillary data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 4 Invalid Non-EOS Ancillary Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting Non-EOS Ancillary data.

Input: Non-EOS Ancillary data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 5 Non-EOS Ancillary Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of Non-EOS Correlative data.

Input: Non-EOS Correlative data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.2 Enhanced Algorithm Integration & Test Thread (T10-02)

5.1.2.1 Thread Objectives

The objectives of the enhanced algorithm integration and test thread are:

- Provide the IR1 Algorithm Integration & Test functionality.
- Provide the functionality of the Algorithm Integration Tools.
- Provide the capability to perform manual Quality Assurance of science data production.

5.1.2.2 Thread Test Description

In order to verify the objectives of the algorithm integration and test thread, the following interface and functional capabilities are required:

- Interface to the SCFs to receive science software delivery packages and return algorithm execution results.
- Interface to configuration management tool, ECS standard compliance tools, and performance measurement tools.

The algorithm integration and test thread re-verifies the IR1 delivered capabilities as well as verifying the TRMM (Rel A) algorithm enhancements. The first sequence of tests are regression tests. A science software delivery package is received into the system, placed under configuration control, and checked for compliance with ECS standards. The algorithm is then compiled, scheduled, and executed. The results are examined and returned to the SCF for verification.

The next sequence verifies the various algorithm related functions delivered with TRMM (Rel A). The tester utilizes the code analysis tool to demonstrate science algorithm code compliance. Code is checked for memory leaks, argument list mismatches, and unused code. Then, the tester runs the science algorithms and monitors their performance. Performance metrics are collected on CPU usage, memory usage, disk space usage, and I/O access counts. Finally, the tester exercises the various quality assessment (QA) functions on algorithm generated products. Both inline and off-line QA is exercised.

Dependencies: (If Applicable)

In order to verify the objectives of the enhanced algorithm integration and test thread, the following interface and functional capabilities are required:

- Ability to download algorithm packages (algorithm, documentation, test results) using ftp, RPC, or media.
- SCF and DAAC versions of the PGS Toolkit.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  - COTS packages to perform standards and compliance checkers.
  - Compiler
  -
- Data:
- Algorithms

5.1.2.2.1 Sequence 1 - Regression Test IR1 ALG I&T

This series of tests verifies that the functionality delivered in the IR1 turnover is still functional. In this sequence, a science software delivery package is received from a SCF and placed under configuration management. The algorithm is checked for compliance with ECS standards. The algorithm is compiled, scheduled and executed. The results are compared with the expected results and sent back to the SCF for verification.

Test Case 1 Successful Science Software Delivery Package

This test verifies that a Science Software Delivery Package can be transferred via ftp from a SCF to a representative DAAC host in the EDF. The Science Software Delivery Package must contain all the required information in order for the algorithm to be placed under configuration control, compiled, and executed. This test verifies that each required part of the package is transferred successfully.

Input: SCF notification of intent to deliver a Science Software Delivery Package.

Output: The entire Science Software Delivery Package to include algorithm identification, algorithm source code, algorithm documentation, list of required inputs, calibration coefficients, processing dependencies, test procedures and test data; appropriate status messages, message log file. This check for completeness is manual and a Delivery Package Evaluation report should be filed to report the completeness of the transfer. The SCF notifies the EDF that the package delivery was completed.

Expected Results: The entire Science Software Delivery Package should be transferred to the requested site, appropriate status messages should be displayed, and the message log file should be updated to reflect the requested Science Software Delivery Package was transferred to the requested location.

Test Case 2 Algorithm Configuration Management

This test verifies that the received Science Software Delivery Package can be placed under configuration management. The configuration management tool must be able to place each element of the transfer package into its system.

Input: Configuration Management tool, the received algorithm package, request to place the received Science Software Delivery Package under configuration management.

Output: Status message, history log updates.

Expected Results: The algorithm should be placed under configuration management, appropriate status message should appear upon completion, the history log should reflect that the package is now under configuration management.

Test Case 3 Algorithm in compliance with ECS standards

This test verifies that the received algorithm is in compliance with ECS standards. The test will utilize a POSIX checker tool to ensure the source code is within ECS guidelines.

Input: Portable Operating System Interface For Computer Environment (POSIX) Checker, algorithm with compliant source code.

Output: Status message, history log.

Expected Results: The POSIX checker should report that the algorithm is within ECS standards.

Test Case 4 Successful compile using SCF version of the PGS toolkit

This test verifies that a received algorithm can be compiled using the SCF version of the PGS toolkit. This is the first test in integrating an algorithm into the ECS system.

Input: SCF version of the PGS toolkit, science algorithm.

Output: Status messages, history log.

Expected Results: The algorithm should be successfully compiled without errors. Appropriate status messages should be generated to reflect this activity and the history log is updated.

Test Case 5 Successful compile using DAAC version of the PGS toolkit

This test verifies that a received algorithm can be ported to the DAAC platform and compiled using the DAAC version of the PGS toolkit.

Input: DAAC version of the PGS toolkit, algorithm to be ported to the DAAC platform.

Output: Status message, history log.

Expected Results: The system should report on the successful porting and compilation of the science algorithm and the history log should be updated.

Test Case 6 Successful scheduling of algorithm

This test verifies that a received algorithm can be scheduled to run using the IR1 scheduler.

Input: IR1 algorithm scheduler, algorithm to be scheduled.

Output: Status message, history log, results of algorithm execution.

Expected Results: The IR1 scheduler should be able to schedule the science algorithm to execute at the DAAC and report on the schedule time, the success of execution, and the results of execution. The history log should be updated to reflect successful scheduling and execution.

Test Case 7 Successful comparison

This test verifies that the expected results provided in the Science Software Delivery Package match those generated during the actual execution of the algorithm.

Input: Comparison tool, expected results, actual results.

Output: Status message, history log.

Expected Results: The system should report on the successful comparison of the expected results with the actual results.

Test Case 8 Successful transfer

This test verifies that the system is able to send the algorithm test results to the SCF for analysis. The results should include: algorithm identification, test times, processor identification, and test results.

Input: Actual test results.

Output: Status message, history log.

Expected Results: The system should report on the successful transfer of the test results to the SCF.

5.1.2.2.2 Sequence 2 - Code Analysis

This series of tests will verify that the code analysis tool performs the required functions for algorithm integration and testing. The sequence will test for memory leaks, argument list mismatches, unused code, and report generation.

Test Case 1 Checking Algorithms for Memory Leaks

This test case will utilize the code analysis tool to check for memory leaks in a received software science package. The tester will run the tool on an algorithm with no memory leaks and an algorithm with known memory leaks. The tool should verify that the one algorithm has no memory leaks as well as find the memory leaks in the corrupt algorithm.

Input: Algorithm with no memory leak, algorithm with known memory leak, code analysis tool.

Output: No memory leaks found and memory leaks found messages.

Expected Results: The code analysis tool should report both conditions to the operator.

Test Case 2 Checking Algorithms for Argument List Mismatches

This test case will utilize the code analysis tool to check for argument list mismatches. The tester will run the tool on an algorithm with no argument list errors as well as an algorithm with type mismatch and an algorithm with number of argument mismatches. The tool should report successful as well as both error conditions found.

Input: Algorithm with no argument mismatch, algorithm with argument type mismatch, algorithm with number of arguments mismatch, and code analysis tool.

Output: No argument mismatch message, Argument type mismatch message, Number of argument mismatch message.

Expected Results: The tool should report all conditions tested for in this test case, namely, no argument mismatch, argument type mismatch, and number of argument mismatch.

Test Case 3 Checking Algorithms for Unused Code

This test case will utilize the code analysis tool to check for unused code. The tester will run the tool on an algorithm with no unused code as well as an algorithm with unused code. The tool should report successful as well as the error condition found.

Input: Algorithm with no unused code, algorithm with unused code, and code analysis tool.

Output: No unused code message, Unused code message.

Expected Results: The tool should report both conditions tested for in this test case, namely, no unused code and unused code.

Test Case 4 Code Analysis Reports

This test case will verify that a report can be generated from the code analysis tool to reflect the conditions tested for in test cases 1 through 3. The generated report must identify the input algorithms as well as the error conditions discovered by the tool.

Input: Test cases 1 through 3, report request.

Output: Code analysis report.

Expected Results: The generated report should identify the input algorithms as well as the error conditions discovered by the tool.

5.1.2.2.3 Sequence 3 - Algorithm Performance Monitoring

This series of tests will monitor algorithm performance. The performance monitoring tool will be tested for CPU time (for both processes and procedures within a process), memory usage, disk space usage, number of I/O accesses, and report generation.

Test Case 1 Algorithm CPU Usage

This test case will run the algorithm performance monitoring tool to determine the algorithm CPU usage. The tool will initiate the algorithm and monitor both the CPU usage of the process as well as any procedures within the process.

Input: Algorithm with procedures within a process, performance monitoring tool.

Output: Message indicating CPU usage of both processes and procedures within the process.

Expected Results: The performance monitoring tool should generate messages indicating the CPU usage of the known processes and procedures with the processes.

Test Case 2 Algorithm Memory Usage - Normal

This test case will run the algorithm performance monitoring tool to determine the memory usage of an algorithm process. The tool will initiate the algorithm and monitor the memory usage.

Input: Algorithm and performance monitoring tool.

Output: Message indicating algorithm memory usage.

Expected Results: The performance monitoring tool should generate messages indicating the memory usage of the algorithm.

Test Case 3 Algorithm Memory Usage - Error

This test case will run the algorithm performance monitoring tool against an algorithm which has been modified to fail to release any memory used during processing once completed. The performance monitoring tool should generate a message indicating that memory was not released back to the system.

Input: Algorithm modified to allocate system memory but not de-allocate any memory, and performance monitoring tool.

Output: Message indicating that the algorithm did not release system memory.

Expected Results: The performance monitoring tool should generate messages indicating that memory was not released back to the system.

Test Case 4 Algorithm Disk Space Usage - Normal

This test case will run the algorithm performance monitoring tool to determine the algorithm disk space usage. The tool will initiate the algorithm and monitor the disk space used by the algorithm.

Input: Algorithm, performance monitoring tool.

Output: Message indicating disk space usage of the algorithm.

Expected Results: The performance monitoring tool should generate messages indicating the disk space usage of the algorithm.

Test Case 5 Algorithm Disk Space Usage - Error

This test case will run the algorithm performance monitoring tool against a modified algorithm which does not release used disk space back to the system. The tool will initiate the algorithm and monitor the disk space used by the algorithm. The tool should report that the disk space used by the algorithm was not released back to the system.

Input: Algorithm modified to not release used disk space, performance monitoring tool.

Output: Message indicating that disk space was not released back to the system.

Expected Results: The performance monitoring tool should generate messages indicating the disk space usage of the algorithm and the fact that the disk space was not released back to the system.

Test Case 6 I/O Access Count

This test case will run the algorithm performance monitoring tool to determine the number of I/O accesses made by the process to each of the input or output data files. The tool will initiate the algorithm and monitor the number of I/O accesses made by the algorithm.

Input: Algorithm, performance monitoring tool.

Output: Message indicating number of I/O accesses made by the algorithm.

Expected Results: The performance monitoring tool should generate messages indicating the number of I/O accesses made by the algorithm to each of the input or output data files used.

Test Case 7 Report Generation

This test case will verify that a report can be generated from the performance monitoring tool to report the measurements made in test cases 1 through 4. The generated report must identify the input algorithms as well as the conditions measured by the tool.

Input: Test cases 1 through 6, report request.

Output: Performance Monitoring Report.

Expected Results: The generated report should identify the input algorithms as well as the measurements made by the tool.

5.1.2.2.4 Sequence 4 Algorithm I&T QA Process

This series of tests will verify that the process for performing inline, off-line, and SCF quality assessment of the algorithms is operational.

Test Case 1 Inline QA Continue

This test case will verify that QA operators can continue algorithm processing after the system suspends execution to await for the QA function to occur. The tester will initiate the algorithm running sequence and wait for the system to issue a pause. When the pause occurs, a message is generated stating that the system is waiting for the QA function to be performed. When this message is received, the operator will elect to perform QA and continue processing. The processing should then continue to the point of advertising the data availability.

Input: Algorithm to execute inline QA, Continue processing command.

Output: QA Pause, advertising of new data products.

Expected Results: The product generation process should pause for QA functions and continue when the continue command is received. The new data product should be advertised upon completion.

Test Case 2 Inline QA Stop

This test case will verify that QA operators can stop algorithm processing after the system issues a wait for QA function pause. The tester will initiate the algorithm running sequence and wait for the product generation to pause. When the execution pauses, a message is generated to tell the operator that the system is waiting for the QA function to be performed. When this message is received, the operator will elect to stop processing. The processing should then terminate and no advertising of the data should occur.

Input: Algorithm to execute inline QA, Stop processing command.

Output: QA Pause, termination of product generation.

Expected Results: The product generation process should pause for QA functions and terminate when the stop command is received. No advertising of new data should occur.

Test Case 3 Off-line QA

This test case will verify that the off-line QA process is functional. The tester will initiate processing with an algorithm which will terminate when is reaches the QA stage. An awaiting QA function message should then go to the DAAC responsible for the QA of the

new product. Once the QA is completed, the inventory attributes and QA summary report are sent to the data server and the data is advertised.

Input: Algorithm to execute off-line QA.

Output: Awaiting QA function message, inventory attributes and QA summary report.

Expected Results: The product generation process should terminate when processing reaches the QA stage. A message should be sent to the responsible DAAC for QA. Once QA is accomplished, the inventory attributes and the QA summary report should be sent to the data server and the data advertised.

Test Case 4 SCF QA

This test case will verify that the SCF QA process is functional. The tester will initiate processing with an algorithm which will terminate when it reaches the QA stage. The processing for this algorithm will send a message to a person responsible for QA coordination at the appropriate SCF. The notification includes information as to the location of the data. The QA function is performed at the SCF. The results of the QA include updated metadata, annotations to the data set, and QA products added to the appropriate Data Server.

Input: Algorithm to execute off-line SCF QA.

Output: Message to the SCF QA responsible person, updated metadata, annotations to the data set, QA products.

Expected Results: The product generation process should terminate when processing reaches the SCF QA stage. A message should be sent to the responsible SCF person for QA. Once QA is accomplished, metadata updates, data set annotations and QA products should be sent to the data server and the data availability advertised.

5.1.3 Product Management Thread (T10-03)

5.1.3.1 Thread Objectives

The product management thread is tested to verify the systems ability to respond to stimuli in the product generation environment. The objectives of the product management thread are:

- Provide the ability to manage the scheduling of resources for product generation.
- Provide the capability to execute data processing requests for product generation.
- Provide the capability to perform simultaneous processing for individual processing requests.

5.1.3.2 Thread Test Description

Testing is performed to demonstrate the allocation of resources for use by a queued request, the actual allocation and staging of resources, the deallocation and de-staging of resources. Testing

also verifies the resolution of resource conflicts scheduled for usage and the reprocessing of a request.

Testing is performed to verify that various hardware resources are defined in the same state for both SDPS and CSMS and that statistical information on hardware resources is collected, maintained, and output in a report format.

Testing will demonstrate that data staging and de-staging are services used to interface with the data server in coordinating the transfer of data to and from the processing CI and the data server. Testing verifies the processing request and the successful transfer of data by the data server. The various data types which can be transferred include PGE scripts, metadata, and algorithm executables, calibration data files, ECS data products and ancillary data products.

Dependencies: (If Applicable)

In order to verify the objectives of the product management thread, the following interface and functional capabilities are required:

- Interface between SDPS resource management and CSMS.
- Data server populated with data sets (i.e., metadata, algorithms, ancillary data, product data).

Test Support Requirements

- Hardware:
  - HP735, SUN Sparc10, DEC 3000/300, IBM RS6000, SGI R3000, and CRAY
- Software:
  - Resource Management COTS tool
  - GUI Interface for queue control and monitoring
  - CSMS LSM Services

    Configuration Management Applications

    Resource Scheduling Application
- Data:
  - Simulated data granules
  - PGE scripts, metadata, algorithm executables, calibration data files, ECS data products and ancillary data products.

5.1.3.2.1 Sequence 1 - Allocating Resource Usage

This sequence of tests verifies that when the data staging initiation check timer has expired, data staging initiation check is performed and resource management information will be updated with the current data processing sub-system's resource characteristics. This sequence of tests will verify processing is performed periodically or the data processing sub-system's resource management information is updated with the current resource usage for both data staging initiation check and PGE execution initiation check. Data staging transfers data from the data

server to the processing configuration item (CI). De-staging transfers the data from the CI to the data server.

Test Case 1 Data Staging Passes Initiation Check

The tester sets the data staging initiation check timer to expire. The tester will then proceed to ensure the resource management information has been updated with the current data processing subsystem resource characteristics. All data required by a PGE must be accessible to the data processing sub-system.

Input: Expiration of data staging initiation check timer. All data required by PGE is accessible.

Output: PGE has successfully passed the data staging initiation check and data staging is initiated with the data server. The resource management information is updated to indicate the reserved disk space needed for data staging. Update the PGE as "Data Staging Initiated", "Data Staging Complete".

Expected Results: Processing log message indicating data staging initiation check passed successfully. Continue processing the next request in the queue.

Test Case 2 Data Staging Fails Initiation Check since PGE data not available

The tester sets the data staging initiation check timer to expire. The tester will then proceed to ensure the resource management information has been updated with the current data processing subsystem resource characteristics. All data required by a PGE is not accessible to the data processing sub-system, and the data processing subsystem's resource characteristics will not support (unable to allocate storage space) the initiation of data staging for a given PGE.

Input: Expiration of data staging initiation check timer. All data required by PGE is not accessible. Tester can take PGE data storage media off-line, making it inaccessible.

Output: The data staging for the PGE does not successfully pass the data staging initiation check and data staging is not initiated with the data server. If a mount request is issued for the storage media, the outcome of this testcase will be dependent upon whether or not a timer pops waiting for the media mount.

Expected Results: Processing log message indicating data staging initiation check failed.

Test Case 3 Data Staging Fails Initiation Check since unable to Allocate Space

The tester sets the data staging initiation check timer to expire. The tester will then proceed to ensure the resource management information has been updated with the current data processing subsystem resource characteristics. All data required by a PGE is accessible to the data processing sub-system, and the data processing subsystem's

resource characteristics will not support (unable to allocate storage space) the initiation of data staging for a given PGE.

Input: Expiration of data staging initiation check timer. All data required by PGE is accessible. Tester may emulate no disk space available by deleting the resources in resource profile using CSMS.

Output: PGE has not successfully passed the data staging initiation check and data staging is not initiated with the data server.

Expected Results: Processing log message indicating data staging initiation check failed.

Test Case 4 PGE Execution Initiation Check

The tester sets the data staging initiation check timer to expire. The tester ensures the resource management information has been updated with the current data processing sub-system resource characteristics. All data required by a PGE is accessible to the data processing sub-system, and the data processing subsystem's resource characteristics supports the initiation of data staging for a given PGE. PGE starts execution.

Input: Expiration of data staging initiation check timer. All data required by PGE is accessible.

Output: PGE has successfully passed the data staging initiation check and is awaiting execution. The resource management information is updated to indicate the reserved disk space needed for data staging. PGE executes successfully. The resource management information is updated to indicate the available disk space needed post PGE execution.

Expected Results: Processing log message indicating data staging initiation check successful, PGE executing, and PGE completes execution.

5.1.3.2.2 Sequence 2 - Allocating Processing Resources

This sequence of tests verifies that when a request for resource allocation is made to support data staging or PGE execution by the process queue service the process resource management service allocates the resource and updates the resource profile. The inputs to the request for allocation are verified. The message log indicates a success/failure. An access request will contain the following information: data identification, data characteristics, resource identification, and type of action (stage, de-Stage, terminate).

Test Case 1 Allocate a Resource for Data Staging

The tester will verify the allocation request of a resource for data staging. The allocation request is provided by the process queue service and contains data, including processing time and disk space.

Input: Allocation access request, valid data for processing.

Output: Algorithm processed in a specified amount of time. A message is written to the processing log indicating a successful allocation.

Expected Results: The resource is allocated to the task for a time < or = the time allotted, and a message is written to the processing log. Resource mode {*on-line*, off-line, available, *unavailable*}

Test Case 2 Allocate a Resource for PGE Execution

The tester will verify the allocation request of a resource for PGE Execution. The allocation request is provided by the process queue service and contains data, including processing time and disk space.

Input: Allocation request, valid data for processing.

Output: Algorithm processed in a specified amount of time. A message is written to the processing log indicating a successful allocation.

Expected Results: The resource is allocated to the task for a time < or = the time allotted, and a message is written to the processing log. Resource mode {*on-line*, off-line, available, *unavailable*}

Test Case 3 Allocate a Preferred Resource

The tester will verify the allocation request of a resource for either staging or PGE execution. The allocation request is provided by the process queue service and contains data, including resource, processing time and disk space. The tester will query the preferred resource to determine if it was selected for processing when resources are in an available state.

Input: Allocation request, valid data for selecting resource and processing algorithm.

Output: Algorithm processed in a specified amount of time. A message is written to the processing log indicating a successful allocation.

Expected Results: The resource is allocated to the task for a time < or = the time allotted, and a message is written to the processing log. Resource mode {*on-line*, off-line, available, *unavailable*}.

Test Case 4 Allocate an Allocated Resource

The tester will verify the allocation request of a resource for processing which is already allocated for either staging or PGE execution. The first allocation request is provided by the process queue service and contains data, including resource, processing time and disk space needed for processing. The tester will ensure a second allocation request, whose processing time and scheduled resource coincide with the first allocated resource, is processed when the resource is made available. To minimize the available resources, the tester can use CSMS to delete resources from the resource profile.

Input: Two processing requests, valid data for selecting resource, the first request must exceed its processing time.

Output: The first request completes processing successfully. A message is written to the processing log indicating a successful deallocation. The second request is processed after the resource has been freed. The same resource is reallocated for the start of the new job.

Expected Results: The resource is allocated to the task for a time < or = the time allotted, and the first and second requests complete successfully.

Test Case 5 Allocate an Undefined Resource

The tester will verify the allocation request of a resource, for either data staging or PGE execution, which is not currently defined to the resource profile. The deallocation request is provided by the process queue service and contains data, including resource information. Tester queries available resources to ensure resource is not defined. The tester can use CSMS to add and delete resources from the resource profile.

Input: Allocation request, resources are not defined in the resource profile.

Output: Resource not allocated to a process, resource is not found in the resource profile. Resource mode {on-line, *off-line*, available, *unavailable*}. A error message is written to the processing log.

Expected Results: Resource is not allocated, and an error message is written to the processing log.

5.1.3.2.3 Sequence 3 - De-Allocating Processing/De-Staging Resources

This sequence of tests verifies that upon completion of a process a request for resource deallocation is made by the process queue service to the process resource management service to deallocate the resource for data de-staging or PGE execution. The inputs to the request for deallocation are verified and updates are made to the resource profile. The message log indicates a success/failure. The stimulus to initiate data de-staging is received as a request from PGE execution management initiated by a completion of PGE execution.

Test Case 1 Deallocate Resource for Data De-Staging

The tester will verify the deallocation request of a resource for data de-staging. The deallocation request is provided by the process queue service and contains data, including resource information.

Input: Deallocation request, valid resource information to deallocate appropriate resource.

Output: Resource deallocated, resource mode {on-line, off-line, *available*, unavailable}. Tester queries available resources. A message is written to the processing log.

Expected Results: Resource deallocated, and a successful completion message is written to the processing log.

Test Case 2 Deallocate Resource for PGE Execution

The tester will verify the deallocation request of a resource for PGE execution. The deallocation request is provided by the process queue service and contains data, including resource information.

Input: Deallocation request, valid resource information to deallocate appropriate resource.

Output: Resource deallocated, tester queries available resources. A message is written to the processing log. Resource mode {on-line, off-line, *available*, unavailable}

Expected Results: Resource deallocated, and a successful completion message is written to the processing log.

Test Case 3 Deallocate Unallocated Resource

The tester will verify the deallocation request for de-staging of PGE execution of a resource which is not currently allocated. The deallocation request is provided by the process queue service and contains data, including resource information. Tester queries available resources to ensure resource is defined and is currently not allocated to a process.

Input: Deallocation request, invalid resource information to deallocate an unallocated resource.

Output: Resource not allocated to a process, resource is not defined in the resource profile. Resource mode {on-line, off-line, *available*, unavailable}. An error message is written to the processing log.

Expected Results: Resource is available, and an error message is written to the processing log.

Test Case 4 Deallocate Undefined Resource

The tester will verify the deallocation request for de-staging or PGE execution of a resource which is not currently defined in the resource profile. The deallocation request is provided by the process queue service and contains data, including resource information. Tester queries available resources to ensure resource is not defined. The tester can utilize CSMS to add and delete resources to the resource profile.

Input: Deallocation request, invalid resource information to deallocate an undefined resource.

Output: Resource not allocated to a process, resource is not found in the resource profile. Resource mode {on-line, off-line, available, *unavailable*}. An error message is written to the processing log.

Expected Results: Resource can not be deallocated, and an error message is written to the processing log.

5.1.3.2.4 Sequence 4 - Resolution of Resource Conflicts

This sequence of tests verifies that a request for resources which result in a conflict is processed by resource management and a resolution is provided.

Test Case 1 Resource Contention

The tester will verify that a staging request, which has a reserved resource allocated for a given start time and completion time, will be re-prioritized if the requested resources are not available. The tester continues to issue new processes to bump the priority of the scheduled staging request.

Input: Process request.

Output: Disk resources scheduled for staging are overwritten and the resource schedule is updated for the bumped staged process request, and the request is re-prioritized and rescheduled. The owner of the request is notified and provided with a new estimated execution time for the bumped request. DADS informs the collocated PGS of any anticipated resource availability conflicts.

Expected Results: Resource rescheduled, and a successful completion message for higher prioritized processes are written to the processing log.

Test Case 2 Reprocess a Request

The tester will verify the system is capable of scheduling and processing a reprocessing request and allocation of resources is performed.

Input: Process first request, issue request for reprocessing.

Output: Disk resources scheduled for staging are rescheduled for the reprocessing request. The owner of the request is notified of the scheduled plan for reprocessing.

Expected Results: Resource rescheduled, and a successful completion message for the first request is written to the log. PGS performs reprocessing according to the PGS reprocessing plan and the availability of resources.

Test Case 3 Reprocess an Invalid Request

The tester will verify the system is capable of rejecting an invalid reprocessing request. No scheduling or processing of the invalid request occurs.

Input: Issue an invalid request for reprocessing.

Output: Reprocessing request is rejected. Event log contains an entry for the rejection of an invalid reprocessing request. Submitter of request notified of rejection of invalid reprocessing request.

Expected Results: Reprocessing request is rejected, no disk resources are scheduled for allocation, and a message is written to the log.

5.1.3.2.5 Sequence 5 - Interface Testing with CSMS Services

This series of tests verifies the functionality of the interface between SDPS and CSMS system management is properly reporting the status of hardware resources. Testing will verify that various hardware resources are defined in the same state for both SDPS and CSMS and that statistical information on hardware resources is collected, maintained, and can be output in a report format.

Test Case 1 — Fatal Hardware Error occurs during Processing

The tester will verify CSMS interface supports the detection of hardware errors. In the event that a PGE is currently processing, CSMS will support the detection of a hardware error, and respond with updates to the resource management information, by marking the availability of a resource.

Input:	Process (PGE) request, simulate/cause a hardware error to occur.
Output:	CSMS will be informed of the problem, log the error condition in the status information file, update the resource management information, and mark the resource unavailable.
Expected Results:	Error condition is logged, and processing of request is terminated. The resource profile is updated, and the resource is marked unavailable.

Test Case 2 — PGE Initiated Unsuccessfully

The tester will verify that when PGE fails to initiate successfully, CSMS will support the detection of the error, and error information will be logged.

Input:	Process request, PGE fails to initiate successfully due to resource faults.
Output:	CSMS will be informed of the problem, log the error condition, and track resource faults.
Expected Results:	Error condition is logged, and processing of request is terminated. The PGE initiation is logged as a failure.

Test Case 3 — PGE reaches Maximum Allowed Execution Time

The tester will verify CSMS interface supports resource monitoring. In the event that a PGE is currently processing and exceeds the maximum allowed or average execution time, CSMS will detect the error in usage of the resource, and the PGE will be terminated.

Input:	Process (PGE) request, PGE execution time exceeds the maximum execution time.
Output:	CSMS will be informed of the problem, log the error condition in the status information file, and the PGE will terminate.

Expected Results: Error condition is logged, and processing of request is terminated. The resource profile is updated and the resource is marked unavailable.

5.1.3.2.6 Sequence 6 - Coordinating Storage Resource Usage

This series of tests verifies the functionality of the interface between SDPS and the data server. Data staging and de-staging are services used to interface with the data server in coordinating the transfer of data to and from the processing CI and the data server. This sequence of test verifies the processing request and the successful transfer of data by the data server. The various data types which can be transferred include PGE scripts, metadata, and algorithm executables, calibration data files, ECS data products and ancillary data products.

Test Case 1 Transfer of Data from Data Server to Processing

The tester will verify the allocation request of a resource for PGE Execution. The allocation request is provided by the process queue service and contains data, including processing time and disk space. All data required by a PGE must be accessible to the data server. No other resources are in contention.

Input: Allocation request, valid data for processing, data exist for data staging.

Output: PGE has successfully passed the data staging initiation check and data staging is initiated with the data server. The resource management information is updated to indicate the reserved disk space needed for data staging, a request is issued for the required data, and the data is transferred from the data server to the processing CI. No resource contention occurs. A message is written to the processing log indicating a successful allocation.

Expected Results: The resource is allocated for a time < or = the time allotted, and a message is written to the processing log. Resource mode {*on-line*, off-line, available, *unavailable*}. The data is staged to the processing CI and the program executable completes. Data is de-staged and the resource is deallocated.

Test Case 2 Transfer of Data from Processing to Data Server

The tester will verify the de-allocation request of a resource for PGE Execution. The de-allocation request is provided by the process queue service. There is no contention for resources.

Input: De-allocation request, valid data for processing, data exists for data de-staging.

Output: Upon completion of algorithm processing, a request is made to de-stage the data. Upon completion of data de-staging, the resource is de-allocated. A message is written to the processing log indicating a successful de-allocation.

Expected Results: Data de-staging completes. Resource mode {*on-line*, off-line, *available*, unavailable}. The data is transferred from the processing CI to the data server.

Test Case 3 Transfer of Data from Processing to Data Server, with two requests for staged data

The tester will verify the completion of PGE execution. The next request queued for processing requires the same data for execution. Instead of a de-allocation request for the required resource and de-staging of the required data, the resources are made available for processing.

Input: Two requests, each request requires the same resources, including data. The requests are prioritized on the same queue.

Output: Upon completion of algorithm processing, a check is made for use of the disk space and data prior to de-allocation, the data is not deleted unless the space is needed. The next queued request is executed. Upon completion of the second request, the data is de-staged, and the resource is de-allocated when a subsequent request requires the space. A message is written to the processing log indicating a successful completion.

Expected Results: Data de-staging completes after successful completion of the second request. Resource mode {*on-line*, off-line, *available*, unavailable}. The data is transferred from the processing CI to the data server.

Test Case 4 Transfer of Data from Processing to Data Server, with Contention for staged data

The tester will verify the completion of PGE execution. A second queue has a request for PGE data and data-staging/processing times of the two requests coincide. One request is selected for processing, and before the de-allocation or de-staging occurs, the second request is processed.

Input: Two requests, each request requires the same resources, including data. The requests are prioritized on different queues.

Output: Upon completion of algorithm processing, a check is made for use of the disk space and data prior to de-allocation, the data is not deleted unless the space is needed. The next queued request is executed. Upon completion of the second request, the data is de-staged, and the resource is de-allocated when a subsequent request requires the space. A message is written to the processing log indicating a successful completion.

Expected Results: Data de-staging completes after successful completion of the second request. Resource mode {*on-line*, off-line, *available*, unavailable}. The data is transferred from the processing CI to the data server.

Test Case 5 Generate a Resource Utilization Report

The tester will verify the ability to generate a resource utilization report.

Input: System actively processing requests. Scheduling and resource allocation for staging and program executables. The administrator issues a command for a request to generate a resource utilization report.

Output: Resource utilization report of the resources available is generated.

Expected Results: Statistics and percent utilization of hardware resources is expected.

Test Case 6 Generate Status Reports

The tester will verify the ability to generate a status report.

Input: Processing system. Scheduling and resource allocation for staging and program executables. The administrator issues command for a request to generate a status report of all queued and currently processing jobs.

Output: Status report of all jobs queued and active.

Expected Results: Statistics and priority of software resources is expected.

5.1.3.2.7 Sequence 7 - GUI Operator Interactive Queries

This sequence of tests verifies that when a query of the resource management list is performed, a determination is made on resource availability. The resource chosen must support the processing of either data staging or PGE execution. PGE execution management services are used to initiate and monitor the execution of a PGE.

Test Case 1 Query for Data Staging

The tester will verify the query of a resource management list for data staging. The resources are defined in the resource profile. The request is provided by the process queue service and contains data, including processing time and disk space. The tester queries available resources to ensure the resource is defined.

Input: Request resource for staging data, request data information includes start and termination time and amount of disk space needed.

Output: Resource management list is queried and a resource with available storage capacity is scheduled. A message is written to the processing log indicating processing has been scheduled for a "standard" request, or staging started for an "on-demand" request.

Expected Results: Resource is scheduled for staging data, and a message is written to the processing log.

Test Case 2 Query for Data Staging when Resource not available

The tester will verify the query of a resource management list for data staging. The resources are defined to the resource profile. The request is provided by the process queue service and contains data, including processing time and disk space. The tester queries available resources to ensure the resource is currently processing another request.

Input: Two requests for resource processing, the first is for data staging processing to be scheduled and performed during a certain time period. The second, is for an "on-demand" request which needs the same resources currently being used for processing (resource unavailable).

Output: Resource management list is queried and a resource with available storage capacity is scheduled for the first request. Two messages written to the processing log indicating processing has been scheduled for a "standard" request, and processing started. The next request will be processed when the resources become available.

Expected Results: Resource is scheduled for staging data, and a message is written to the processing log indicating when the second resource will be processed.

Test Case 3 Query for Product Generation Executable

The tester will verify the query of a resource, for PGE execution, which is currently defined in the resource profile. The request is provided by the process queue service and contains data, including resource information. Tester queries available resources to ensure resource is defined. The tester can utilize CSMS to add and delete resources to the resource profile.

Input: Request resource for PGE execution, including input data identification and source, output data identification and source, PGE identification, priority assigned for execution, and resources required.

Output: Resource with available storage capacity is scheduled. Message is written to the processing log.

Expected Results: Resource is scheduled for PGE execution, and a message is written to the processing log.

Test Case 4 Query for PGE when Resource not available

The tester will verify the query of a resource management list for PGE execution. The resources are defined in the resource profile. The request is provided by the process queue service and contains data, including processing time and disk space. The tester queries available resources to ensure the resource is currently processing another request.

Input: Two requests for resource processing, including input data identification and source, output data identification and source,

PGE identification, priority assigned for execution, and resources required. The first, science software PGE, is for PGE execution processing to be scheduled and performed during a certain time period. The second, is for a periodically performed processing job such as resource maintenance request, which needs the same resources currently being used for processing (resource unavailable).

Output: Resource management list is queried and a resource with available storage capacity is scheduled for the first request. A message is written to the processing log indicating processing has been scheduled for a "standard" request, and processing started. The next request will be processed when resources become available.

Expected Results: Resource is scheduled for execution, and a message is written to the processing log indicating when the second resource will be processed.

5.1.4 Enhanced System Access Thread (T10-04)

5.1.4 .1 Thread Objectives

The objectives of the enhanced system access thread are as follows:

- Regression test IR1 system access functionality.
- Provide management services.
- Provide communication services.

5.1.4 .2 Thread Test Description

Testing verifies IR1 system access functionality (i.e.,, logon/logoff, transfer of data files, access to e-mail and bulletin board services, and V0 network accessibility).

Testing verifies the functionality of management services. The use of office automation tools to maintain a training schedule, a site inventory of spare and consumable parts, and establishment of site policies and procedures is verified. The ability to detect, isolate, identify, and resolve system faults at the host access level is verified. The ability to monitor and collect statistical data for a LAN is verified.

The capability to communicate in a network environment which supports various protocols (TCP/IP, RPC, etc) is verified.

Dependencies: (If Applicable)

In order to verify the objectives of the enhanced system access thread, the following interface and functional capabilities are required:

- LAN
- WAN

Test Support Requirements

- Hardware:
  - TBS
- Software:
  - TBS
- Data:

5.1.4.2.1 Sequence 1 - Logon/Logoff

The following series of tests verify the ability to properly logon/logoff the system (local and remote). Both valid and invalid testing will be conducted.

Test Case 1 — Local Logon - Rlogin

This test verifies that a tester is able to securely logon to another local host. All activity for each user account is recorded in the history log file.

Input: Valid login id. Rlogin to another local platform. Logoff. Network analyzer.

Output: Event log contains an entry for the login on the local workstation, the remote login to the local platform, and the logoff from the local platform. Network analyzer validates that the port connection to the local platform was successfully terminated/relinquished when the logoff was executed.

Expected Results: Successful rlogin to remote platform. Port connection closed after logoff.

Test Case 2 — Remote Logon - Telnet

This test verifies that a tester is able to securely logon to another host on a WAN. All activity for each user account is recorded in the history log file.

Input: Valid login id. Telnet to remote platform on the WAN. Logoff. Network analyzer.

Output: Event log contains an entry for the login on the local workstation, the telnet login to the remote platform, and the logoff from the remote platform. Network analyzer validates that the port connection to the remote platform was successfully terminated/relinquished when the logoff was executed.

Expected Results: Successful telnet to remote platform. Port connection closed after logoff.

Test Case 3 Abnormal Network Termination

This test verifies that the port connection is relinquished/terminated when a user is abnormally terminated from a network connection (i.e.,, system halt during network access). This test is executed for both a LAN and WAN connection.

Input: Valid login id. Login to local workstation and then login to remote platform (rlogin or telnet). Halt the local workstation. Network analyzer.

Output: Event log contains an entry for the login on the local workstation and the login to the remote platform. Network analyzer validates that the port connection to the remote platform was successfully terminated/relinquished when the local workstation was halted.

Expected Results: Successful port termination when an abnormal logoff condition occurs.

Test Case 4 System Accessibility

This test verifies that users are provided, at all times, with informational status messages regarding system accessibility.

Input: User logon to ECS system. Status messages of current system service accessibility, including a list of services currently unavailable and the predicted time of resumption of service.

Output: Event log contains an entry for user logon. User local workstation contains current status of ECS service accessibility.

Expected Results: User logon successful. User receives status of availability ECS system services and predicted time of resumption for services currently inoperable.

5.1.4.2.2 Sequence 2 - Transfer Data Files

The following series of tests verify the ability to perform file transfers within a LAN and WAN network environment. File sizes used for test purposes will vary to verify possible network limitations on file size transfers.

Test Case 1 LAN Transfer - File less than 5 Meg

This test verifies the capability to perform a local file transfer of a file less than 5 meg in size. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, LAN, data file less than 5 meg in size, ftp or RPC software package.

Output: File transfer completed. Complete data file received on local remote platform. Event log contains an entry for the login and transfer of the data file.

Expected Results: Data file received on local remote platform is the same as the data file on the local workstation.

Test Case 2 WAN FTP - File less than 5 Meg

This test verifies the capability to perform a remote file transfer of a file less than 5 meg in size. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, WAN, data file less than 5 meg in size, ftp or RPC software package.

Output: File transfer completed. Complete data file received on remote platform. Event log contains an entry for the login and transfer of the data file.

Expected Results: Data file received on remote platform is the same as the data file on the local workstation.

Test Case 3 LAN FTP - File larger than 5 Meg

This test verifies the capability to perform a local file transfer of a file larger than 5 meg in size. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, LAN, data file larger than 5 meg in size, ftp or RPC software package.

Output: File transfer completed. Complete data file received on local remote platform. Event log contains an entry for the login and transfer of the data file.

Expected Results: Data file received on local remote platform is the same as the data file on the local workstation.

Test Case 4 WAN FTP - File larger than 5 Meg

This test verifies the capability to perform a remote file transfer of a file larger than 5 meg in size. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, WAN, data file larger than 5 meg in size, ftp or RPC software package.

Output: File transfer completed. Complete data file received on remote platform. Event log contains an entry for the login and transfer of the data file.

Expected Results: Data file received on remote platform is the same as the data file on the local workstation.

Test Case 5 Pull Data File - File less than 5 Meg

This test verifies the capability to perform a file transfer of a file less than 5 meg in size from a remote platform to the local workstation. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, WAN, data file less than 5 meg in size on a remote network, RPC or ftp software package.

Output: File transfer from the remote network to the local workstation completed. Complete data file received on the local workstation. Event log contains an entry for the login and receipt of the data file.

Expected Results: Data file received on the local workstation is the same as the data file on the remote platform.

Test Case 6 Pull Data File - File larger than 5 Meg

This test verifies the capability to perform a file transfer of a file larger than 5 meg in size from a remote platform to the local workstation. This test case is executed for both an ftp and RPC file transfer.

Input: Valid login, WAN, data file larger than 5 meg in size on a remote network, RPC or ftp software package.

Output: File transfer from the remote network to the local workstation completed. Complete data file received on the local workstation. Event log contains an entry for the login and receipt of the data file.

Expected Results: Data file received on the local workstation is the same as the data file on the remote platform.

5.1.4.2.3 Sequence 3 - Network Communication

The following series of tests verify the ability to send e-mail in a LAN and WAN network environment. The ability to post articles to the bulletin board for access by all authorized users is verified. Testing will be performed to verify that a user (client) can not delete articles from the bulletin board. Testing will verify that only the administrator of the bulletin board can delete articles from the bulletin board.

Test Case 1 E-mail Message Across the LAN

This test verifies that a tester, using a valid account, is able to e-mail a message to other accounts at the same site via the Ethernet LAN. This test is to be performed on each site's LAN.

Input: Login as a valid user and generate an e-mail message to another known registered user on the same LAN. Registered user address information is available through an address book (listing). Send message.

Output: Message is received at the receiving address. Recipient is able to access and read the message.

Expected Results: Message is sent to the recipient. Recipient is able to open the e-mail to read it. The history log contains an entry if any error messages were encountered during the transfer.

Test Case 2 LAN E-mail Message - Invalid Addressee

This test verifies that a tester, using a valid account, receives an error message when attempting to send an e-mail message to a registered user with an improper address defined.

Input: Login as a valid user and generate an e-mail message to another known registered user, inserting an incorrect address or mis-spelled name. Send message.

Output: Sender should receive a message indicating that the message was un-deliverable because of an incorrect address or addressee.

Expected Results: Message transmission fails and sender is notified of the failure. The history log should contain an entry of any error messages encountered during the transfer.

Test Case 3 E-mail Message across the WAN

This test verifies that a tester using valid accounts is able to an send e-mail messages across a WAN. This test is repeated for each WAN supported in Rel A.

Input: Login as a valid user and generate an e-mail message to another known registered user on the WAN under test. Registered user address information is available through an address book (listing). Send message.

Output: Message is received at the receiving address. Recipient is able to access and read the message.

Expected Results: Message is sent to the recipient. Recipient is able to open the e-mail to read it. The history log contains an entry if any error messages were encountered during the transfer.

Test Case 4 WAN E-mail Message - Invalid Addressee

This test verifies that a tester, using a valid account, receives an error message when attempting to send an e-mail message to a registered user with an improper address defined.

Input: Login as a valid user and generate an e-mail message to another known registered user, inserting an incorrect address or mis-spelled name. Send message.

Output: Sender should receive a message indicating that the message was un-deliverable because of an incorrect address or addressee.

Expected Results: Message transmission fails and sender is notified of the failure. The history log should contain an entry of any error messages encountered during the transfer.

Test Case 5 Posting Articles to the BBS

This test verifies the ability to post articles to the BBS.

Input: Article item to post to the Bulletin Board.

Output: Article item is successfully posted to the BB.

Expected Results: Article item posted to the BB and is accessible to users that have access to that BB.

Test Case 6 Retrieving Articles from the BBS

This test verifies the ability to retrieve articles from the BBS.

Input: Select the bulletin board that contains item to be retrieved. Retrieve item and save it to a local working directory.

Output: The selected bulletin board is displayed, the article is retrieved and is now available in the local working directory.

Expected Results: User has a local working copy of item retrieved from the BB.

Test Case 7 User Deletion of Bulletin Board Articles

This test verifies that a user is prohibited from deleting articles from the bulletin board. This test case is executed to verify that both an article posted by the client and an article not posted by the client can not be deleted by the client.

Input: Select the bulletin board item to delete.

Output: Item not deleted from the bulletin board. User receives a message indicating that articles can only be deleted from the bulletin board by and authorized administrator of the bulletin board.

Expected Results: Item not deleted. Clients are not granted access to delete bulletin board items because of the inability to remove the item from all bulletin board servers in a multi server environment.

Test Case 8 Administrator Deletion of Bulletin Board Articles

This test verifies that only the administrator of the bulletin board can delete items from the bulletin board.

Input: Administrator tool to maintain bulletin board. Selected item to delete from the bulletin board.

Output: The article is deleted from all bulletin board servers in the multi server environment. The item is no longer accessible by any client.

Expected Results: The article is removed from all servers within the multi server environment.

5.1.4.2.4 Sequence 4 - V0 Network Access

The following series of tests verify the V0 network connectivity used in IR1. All tests will be performed on host machines within each site (GSFC, MSFC, LaRC, and EDF) that are directly connected to the V0 network.

Test Case 1 EDF to GSFC V0 Access

This test verifies that the dedicated V0 link between GSFC and EDF is operational.

Input: Valid login/password for dedicated V0 host machine at GSFC.

Output: Successful logon to V0 host at GSFC. Event log contains an entry for the login to the V0 host at GSFC.

Expected Results: Connectivity to GSFC V0 platform successful.

Test Case 2 EDF to LaRC V0 Access

This test verifies that the dedicated V0 link between LaRC and EDF is operational.

Input: Valid login/password for dedicated V0 host machine at LaRC.

Output: Successful logon to V0 host at LaRC. Event log contains an entry for the login to the V0 host at LaRC.

Expected Results: Connectivity to LaRC V0 platform successful.

Test Case 3 EDF to MSFC V0 Access

This test verifies that the dedicated V0 link between MSFC and EDF is operational.

Input: Valid login/password for dedicated V0 host machine at MSFC.

Output: Successful logon to V0 host at MSFC. Event log contains an entry for the login to the V0 host at MSFC.

Expected Results: Connectivity to MSFC V0 platform successful.

5.1.4.2.5 Sequence 5 - Use of Office Automation Tools

The following series of tests verify the capability to access various office automation (OA) tools via the desktop to maintain a training schedule, an inventory of spare and consumable parts, and establish site policies and procedures. The capability to activate a graphical interface tool for generation of graphical displays/diagrams is demonstrated. Testing will verify the ability for OA tools to connect to a modem for electronic transfer of OA generated data.

Test Case 1 Spreadsheet Tool

This test verifies the ability to access a spreadsheet tool to maintain a training schedule and site inventory data.

Input: Desktop, icon to allow selection to activate spreadsheet application.

Output: Spreadsheet application activated. Event log contains an entry for the activation of the spreadsheet application.

Expected Results: User able to activate a spreadsheet tool from the desktop.

Test Case 2 Word Processing Tool

This test verifies the ability to access a word processing tool to establish/maintain site policies and procedures.

Input: Desktop, icon to allow selection to activate word processing application.

Output: Word processing application activated. Event log contains an entry for the activation of the word processing application.

Expected Results: User able to activate a word processing tool from the desktop.

Test Case 3 Graphical Interface Tool

This test verifies the ability to access a graphical interface tool.

Input: Desktop, icon to allow selection to activate graphical interface application.

Output: Graphical interface application activated. Event log contains an entry for the activation of the graphical interface application.

Expected Results: User able to activate a graphical interface tool from the desktop.

Test Case 4 OA Tool Interface to a Modem

This test verifies the ability to access a modem from an OA tool to support electronic transfer of OA generated data.

Input: Desktop, icon to allow selection to activate a modem interface. User selection of OA generated data to transfer electronically to remote location.

Output: Modem interface activated. Selected data transferred to remote location. Event log contains an entry for the activation of the modem interface and the transfer of the data.

Expected Results: User able to transfer OA generated data electronically to a remote location.

Test Case 5 Established Policies and Procedures

This test verifies that policies and procedures have been established to support network communications and the use of OA tools for generation of policies and procedures.

Input: Existing policies and procedures.

Output: Inspection verifies existence of required policies and procedures.

Expected Results: Policies and procedures have been established in accordance with the Level 3 F&PRS requirements.

5.1.4.2.6 Sequence 6 - Accessing Training Materials

The following series of tests verify the ability to retrieve training information via E-mail and bulletin board service (BBS).

Test Case 1 Receive Training Material via E-mail

This test verifies the ability to retrieve training information via E-mail.

Input: Training material issued via E-mail to known system user. User logon and access E-mail.

Output: User receives notification of existing unread E-mail when logon is performed. User able to receive and open training material received via E-mail.

Expected Results: User notified of E-mail and able to open and read mail contents.

Test Case 2 Access Training Material via BBS

This test verifies the ability to retrieve training information via BBS.

Input: User access to bulletin board service. Existing 'Training' bulletin board and articles posted to 'Training' bulletin board.

Output: User is able to access training materials posted to the 'Training' bulletin board.

Expected Results: User is able to access articles posted on the 'Training' bulletin board.

5.1.4.2.7 Sequence 7 - Statistical Data Collection

The following series of tests verify the collection of statistical data to support report generation capabilities.

Test Case 1 Event Logs

This test verifies that statistical data is being recorded in the event log.

Input: Event log.

Output: Review of the event log.

Expected Results: Event log is recording statistical data.

Test Case 2 History Logs

This test verifies that statistical data is being recorded in the history log.

Input: History logs.

Output: Review of the history log.

Expected Results: History log is recording statistical data.

5.1.5 CSMS Security Thread (T10-05)

5.1.5.1 Thread Objectives

The objectives of the CSMS security thread are:

- Provide the ability to regulate access to networked resources based on names and group membership privileges. (Authorization)
- Provide the ability to prove authenticity in a client/server environment before access is granted for a resource. (Authentication)
- Provide the ability to perform security auditing and generate reports for auditing purposes.
- Provide the ability to perform security risk processing and compromise.

5.1.5.2 Thread Test Description

Testing is performed to verify authorized uses of the system, and prevent unauthorized use of resources by allowing the administrator to define the principals or domain, and the permissions in the ACLs associated with each resource. To ensure the security criteria adheres to the C2 (Controlled Access Protection) as documented by the Department of Defense, security directives, parameters, and thresholds for the generation of alerts are established in the policies and procedures application service. Testing is performed to verify C2 criteria is satisfied and the interfaces within the security thread are operational (i.e., policies and procedures application service interface with the security application service).

Network security or authentication for access to resources in a client/server environment is verified by validating a users authorization to use the system, make a request to a server in the network, authenticate the server, and in turn, the server authenticates the client prior to resources being accessed. The security of sending and receiving the data is made possible by the unique key that is agreed upon by the sender and receiver.

Testing is performed to verify the ability to counter measure security violations for data communications, and generate reports related to security events.

Dependencies: (If Applicable)

In order to verify the objectives of the CSMS security thread, the following interface and functional capabilities are required:

- Accounting and accountability services must be established.
- Security directives and thresholds to support generation of alerts must be established.
- Report generation application service must be established.

Test Support Requirements

- Hardware:
  - DCE workstation
- Software:
  - emulation for remote GSFC site
- Data:
  - simulated data granules

5.1.5.2.1 Sequence 1 - Discretionary Access Control

This sequence of tests verifies the ability to establish security directives, parameters, and thresholds for the generation of alerts. Testing verifies the interface to the policies and procedures application service. Testing verifies the ability to set passwords, password auditing, define access controls, and access/privileges checking. Account authorizations are dependent on NASA defined policies and procedures. The ECS systems security grants access through user classes. Guest accounts have limited access to data while registered user accounts have greater access to data. The valid access control operations for a file resource are: read, write, execute. The task of defining and setting access controls is the responsibility of the application programmer/administrator developing the server.

Test Case 1 Set Password Guidelines

The tester sets the password guidelines to establish directives for password control via the policy procedures management service. The password guidelines should not violate standard password conventions such as special characters, numeric type, and length of password.

Input: Set password guidelines command and parameters. Password guidelines are used to insure strict adherence to password conventions as defined using policy and procedures management services.

Output: Password guidelines are set and are reasonable.

Expected Results: Password guidelines are set and are in accordance with the documented C2 standards.

Test Case 2 Define User Classes - Guest

The tester sets the user classes to establish parameters for user control via policy and procedures management service. Security covers two general principals; resources, and users who want to modify or access resources. Classes are defined to specify user-to-permission authority via a class which specifies what is accessible to a user group, and the valid access controls for resources which are available to the user group.

Input: Define user classes command and parameters. Define four user classes; guest data access (non-public), guest data access (public), guest data access (non-public with NASA privileges), and guest

data access (public with NASA privileges). Hence, each class will be used in later test for setting user-to-permission relationships.

Output: User classes are defined, and the access control list identifies each set user class.

Expected Results: User classes are defined for a guest user.

Test Case 3 Define User Classes - Registered

The tester sets the user classes to establish parameters for user control via policy and procedures management service. Security covers two general principals; resources, and users who want to modify or access resources. Classes are defined to specify user-to-permission authority via a class which specifies what is accessible to a user group, and the valid access controls for resources which are available to the user group.

Input: Define user classes command and parameters. Define two user classes; registered user data access (non-public), registered user data access (public) . Hence, each class will be used in later test for setting user-to-permission relationships.

Output: User classes are defined, and the access control list identifies each set user class.

Expected Results: User classes are defined for a registered user.

Test Case 4 Define Access Controls - Read

The tester defines the access controls directive to specify class-to-permission relationships. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. The access operations for a file resource are: read, write, and execute.

Input: Define access control command and parameters. Define class access permission (*read*, write, execute).

Output: Class has (*read*, write, execute) permission for a file resource.

Expected Results: Class has the correct permission, and the ACL list indicates the set class.

Test Case 5 Define Access Controls - Write

The tester sets the access control directives to specify class-to-permission relationships. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. These access operations for a file resource are: read, write, and execute.

Input: Define access control command and parameters. Define class access permission (read, *write*, execute).

Output: Class has (read, *write*, execute) permission for a file resource.

Expected Results: Class has the correct permission, and the ACL list indicates the set class.

Test Case 6 Define Access Controls - Execute

The tester sets the access control directives to specify class-to-permission relationships. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. These access operations for a file resource are: read, write, and execute.

Input: Define access control command and parameters. Define class access permission (read, write, *execute*).

Output: Class has (read, write, *execute*) permission for a file resource.

Expected Results: Class has the correct permission, and the ACL list indicates the set class.

Test Case 7 Define Access Controls - Invalid

The tester sets the access controls to specify an invalid class-to-permission relationship. The tester sets an invalid access control for the class, and the command is rejected.

Input: Define access control command and parameters. Class identifier, the user sets an invalid access permission (i.e., none).

Output: An error message is issued to indicate the selected access control operation does not exist.

Expected Results: An error message is logged, the command to set the access control to an invalid value is rejected.

Test Case 8 Define Security Events

The tester defines the security events to be monitored by ECS via the policy and procedures management application service. The security events are a set of ECS defined events which are included as parameters when specifying the define security events command.

Output: Define security event command and parameters to monitor data access to sensitive and public file resources.

Output: Security event is defined successfully.

Expected Results: Any security accesses not authorized in the future for sensitive files will be logged, and any general access to public domain resources will be logged in an audit trail.

Test Case 9 Define Recovery Procedures

The tester will verify the ability to define the recovery procedure corresponding to the defined security events.

Input: Define the recovery procedure which corresponds to the security event to monitor data access.

Output: Recovery procedure is defined successfully.

Expected Results: Recovery procedure, set corresponding to defined security event, is defined successfully.

Test Case 10 Set Password

The tester will verify the ability to change a set password via the LSM security application service. Once the set password guidelines are defined to the system, a authorized user is allowed to reset/change a password according to the specified guidelines.

Input: Set password command and parameter. The user has authority to perform set password for administration login authority.

Output: The password is changed. Audit information is logged about the access attempt and password change. Audit information will log the event, and not the details such as the new password (this would be a security compromise).

Expected Results: The administrator has authority to perform administration functions via login using the new password.

Test Case 11 Set Invalid Password

The tester will verify the ability to set an invalid password, and receive an alert according to the policies and procedures set by the defined password guidelines.

Input: Set password command and parameters. Password guidelines indicate a password is equal to eight characters in length. Administrator logs in, and changes the password to a password nine characters in length.

Output: The password is not changed due to the guidelines established. The event is logged, and an alert is generated to notify the user.

Expected Results: The alert is generated since the attempt to change the password is not in accordance with the password guidelines. The event is recorded in the audit trail.

Test Case 12 Set Access Controls - Read

The tester will verify the ability to set the access controls to specify user-to-permission/class relationships for a given resource and establishes the ACL file. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. These access operations for a file resource are: read, write, and execute. Setting access controls is an LSM security application service.

Input: Set access controls command and parameters. Set user permission according to a class with access permission (*read*, write, execute).

Output: User has (*read*, write, execute) permission for a file resource in accordance with the defined class.

Expected Results: User has the correct permission, and the ACL list indicates the set permission.

Test Case 13 Set Access Controls - Write

The tester will verify the ability to set the access controls to specify user-to-permission/class relationships for a given resource and establishes the ACL file. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. Setting access controls is an LSM security application service.

Input: Set access controls command and parameters. Set user permission according to a class with access permission (read, *write*, execute).

Output: User has (read, *write*, execute) permission for a file resource in accordance with the defined class.

Expected Results: User has the correct permission, and the ACL list indicates the set permission.

Test Case 14 Set Access Controls - Execute

The tester will verify the ability to set the access controls to specify user-to-permission/class relationships for a given resource and establishes the ACL file. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. Setting access controls is an LSM security application service.

Input: Set access controls command and parameters. Set user permission according to a class with access permission (read, write, *execute*).

Output: User has (read, write, *execute*) permission for a file resource in accordance with the defined class.

Expected Results: User has the correct permission, and the ACL list indicates the set permission.

Test Case 15 Update Access Controls - from Read to Write

The tester will verify the ability to change the access controls to specify a user-to-permission/class relationship for a given resource and establish the ACL file. An access control list (ACL) grants permission to different kinds of users to perform different kinds of access operations. These access operations for a file resource are: read, write, and execute. The task of modifying the ACL is performed by the ECS systems administrator role.

Input: User invokes an application to edit the ACLs associated with that service. Set access controls command and parameters. Change the user permission by updating the class permission from read to write access.

Output: A message is communicated to the server to get the ACL and let the administrator edit it. The modified ACL list is sent back to the server. The server saves it on the disk and updates the copy in memory. User has (read, *write*, execute) permission for a file resource in accordance with the defined class.

Expected Results: User has the correct permission, and the ACL list indicates the changes access control from read to write.

5.1.5.2.2 Sequence 2 - Validating Access/Privileges

This sequence of tests verifies the countermeasures for security violations to data communications. A terminal will be designated as a security alarm console where all audible events are displayed in accordance with the defined security events. Security covers two general principals; resources, and users who want to modify or access resources. Testing verifies the ability of a user to register for privileges within the ECS system and access data according to the defined privileges. A request for sensitive or restricted resources by an unauthorized user will be displayed, and request for local resources by an authorized user is verified by DCE.

Test Case 1 Verify Access Control List (ACL)

The tester verifies the ability to set the access controls to specify user-to-permission relationships for a given resource and establishes the ACL file. The tester list the contents of the access control list.

Input: Define user classes, define access controls, and set access controls. See the above test cases requiring updates to the ACL list.

Output: The access control file entries are valid.

Expected Results: The access control list is a valid and complete listing of all the user/class-to-permission relationships.

Test Case 2 User is Unauthorized - Update ACL

The tester verifies the ability to verify a user is denied access authority to update the access control list when not authorized to update the ACL list.

Input: Login with an invalid password.

Output: The user access authority is denied.

Expected Results: The user access authority is denied, the user is notified via the DCE and an alert is generated at the console. The event is recorded in the audit trail.

Test Case 3 User Attempts to Access Sensitive Files

The tester verifies that when confidential information is advertised, attempts to access the information are recorded. An audit trail, a component of network security, will display a valid, authorized account attempting to access a sensitive file. The tester can verify access of non-public data by a guest user, and access of non-public data by a registered user with public only authority.

Input: Account, advertisement, and sensitive data. Accessing privileges for the class associated to the sensitive data are restricted.

Output: The user access authority is denied, and a security alert is generated at the security console. The user is notified that the advertised sensitive data can not be accessed without BYPASS privileges or updated privileges.

Expected Results: The user access authority is denied, and an audit trail records invalid access attempts. The advertisement should specify the authorization needed to access the data. The administrator may grant BYPASS privilege to the user.

Test Case 4 User Repeatedly Attempts to Access Sensitive Files

The tester verifies that when confidential information is advertised, attempts to access the information are recorded. An audit trail, a component of network security, will display a valid, authorized account attempting to access a sensitive file. After a predetermined number of unsuccessful attempts to access data, the user is disconnected.

Input: Account, advertisement, and sensitive data. Accessing privileges for the class associated to the sensitive data are restricted, and the user Dow not have authority to access such data. The user attempts to access sensitive data repeatedly.

Output: The user access authority is denied, and a security alert is generated at the security console. The user is notified that the advertised sensitive data can not be accessed without BYPASS privileges or updated privileges. After a predetermined number of attempts the user is disconnected from the system.

Expected Results: The user access authority is denied, and an audit trail records invalid access attempts. The advertisement should specify the authorization needed to access the data. The administrator may grant BYPASS privilege to the user, but in this instance, after a predetermined number of attempts, the user is disconnected from the system.

Test Case 5 User Registration through E-mail - Public

The tester verifies the ability of a user to register through E-mail for privileges within the ECS system.

Input: E-mail message requesting registration application for ECS services.

Output: ECS service application sent to requesting user. Application is returned and processed . Registered user now has access to public data volumes.

Expected Results: The user access authority to public data volumes is granted. The user can test access authority by requesting public data, as a registered user. User activities are logged.

Test Case 6 User Registration through BBs services - Non-Public

The tester verifies the ability of a user to register through BBs services for privileges within the ECS system.

Input: BBs link and complete the application requesting registration to access data within non-public data access privileges (data privilege is browse).

Output: ECS service application received and processed. Application is processed . Registered user now has access to browse non-public data volumes.

Expected Results: The user access authority to non-public data volumes is granted. The user can test access authority by requesting non-public data, as a registered user. The user can browse public and non-public data. User access and activities are logged.

Test Case 7 User Registration through Internet Interfaces

The tester verifies the ability of a user to register through the guest Internet interface for privileges within the ECS system.

Input: Guest login and password. Registration application request.

Output: ECS service application received and processed. Application is processed . User now has access to the ECS system.

Expected Results: The user access authority to public data volumes is granted. The user can test access authority by requesting public data, as a registered user. The user can browse public data. User access and activities are logged.

Test Case 8 User Registration - Existing Profile

The tester verifies the ability of a registered user to access the directory and modify an existing profile and obtain higher privileges than a user without an existing profile. Successful downloading of the scientist workbench to the local register's workstation is required.

Input: Registered user login, obtain and complete the on-line registration application request. The tester should define and modify the following user profile information to include: data distribution address and media, default query parameters, user electronic address, user expertise level, terminal characteristics, and technical specialty. All other information is protected by access control.

Output: ECS service application received and processed. Application is processed . User now has higher privileges to the ECS system.

Expected Results: A higher user access authority is granted. The user can test access authority by requesting data. User access and activities are logged.

Test Case 9 Object Reuse

The tester verifies the ability of the system to revoke all authorizations to information prior to assignment, allocation, or reallocation to another user. This test will verify that a prior users access to an object has been released back to the system, and no subsequent user will inherit prior access privileges.

Input: Registered user login, perform an on-demand request to query restricted data. Next, a guest user attempts to gain query access to the same data which the guest user does not have access to.

Output: ECS service application received and processed. Application is processed . The registered user is able to access the ECS system and perform the query. The guest user is not able to query the restricted data.

Expected Results: Access authority is granted to the registered user, and is not granted to the guest user. User access and activities are logged.

5.1.5.2.3 Sequence 3 - Defining Network Security

This sequence of tests verifies that each server in a networked environment defines the type of permissions to access its resources and defines the permissions in the access control list for the service/operations it provides.

Test Case 1 Perform steps in Sequence 1 for a variety of data servers offering unique services at EDF and an emulated GSFC facility

5.1.5.2.4 Sequence 4 - Network Security

This sequence of test verifies that general resources (resources other than files i.e., records, databases, servers, and directories), and file resources in a networked environment are protected. The hand shake or disclosure of authentication information occurring between client and server for resource access is referred to as authentication.

Test Case 1 Verify Compliance to Security Directives across Network

The tester verifies the security events, and access controls are complied to when accessing and sending data across the network and the disclosure of authentication information over the network. The tester verifies that network logins are performed for a valid user class at EDF and GSFC. The two facilities have remote capabilities of copying files.

Input: The security access controls are set, and the security events are defined as specified in the above test cases. A request is made from EDF to copy a file down to GSFC. The user has access authority to copy the data file to the remote GSFC facility, DCE authorization and database authorization. The user specifies the desired node.

Output: User authorization and authentication over the network are validated, and the file is transferred from EDF to GSFC. The data to be exchanged over the network is encrypted to make eavesdropping unproductive.

Expected Results: The user will send a version of the file which will be placed in receiver's/target home directory. The data file transferred will be identical to the data file received at GSFC.

Test Case 2 Verify ACL Editing Permission

The tester verifies the security events, and access controls are complied to when accessing ACL information. The tester verifies that a network login is performed for a valid user class at EDF and the user has authority to the edit the ACL list.

Input: Set access control commands and parameters. The security access controls are set, and the security events are defined which permits a user to have access control list authorization.

Output: User authorization and authentication are validated, and the user is able to change the access control list.

Expected Results: The user updates the ACL list by adding an additional user class. A message is logged as a result of accessing and performing the ACL transaction. The event is recorded in the audit trail.

Test Case 3 Verify Database Permission

The tester verifies the security events, and access controls are complied to when accessing database information. The tester verifies that a network login is performed for a valid user class at EDF which allows the user to have write authority to the database. The user attempts to perform database cleanup to delete unwanted or expired data from storage media.

Input: The security access controls are set, and the security events are defined which permits a user to have database administration authority. A request is made from the EDF DBA to delete entries in the database and perform cleanup.

Output: User authorization and authentication are validated, and the user is able to perform database cleanup.

Expected Results: The user performs database cleanup by deleting entries in the database. A message is logged as a result of accessing and performing the database transaction.

Test Case 4 Verify Configuration Management Permission

The tester verifies the authority to access/modify configuration management information within the configuration management system. In order not to affect current configuration models, the tester should provide a dummy configuration file for testing purposes or run this test during setup.

Input: Valid login as a CM user. Check-out an existing file within the CM system, modify the header of the configuration file. Perform a check-in, to transfer the updated model back to the configuration management system.

Output: User authorization and authentication are validated, and the user is able to update the configuration management file.

Expected Results: The file is successfully updated and checked back into the CM system. An audit trail message is logged as a result of accessing and performing the CM transaction.

Test Case 5 Verify Report Generation

The tester verifies the ability to generate report via information provided by accounting and accountability application service. The tester can verify report generation for data accesses and activities via the report generation application service.

Input: Generate reports command and parameters. The security access controls are set, and the security events are defined which permits a user who has authority to generate accounting information and user activities information.

Output: User activity reports and file access reports are generated.

Expected Results: The authorized user generates a report of all accesses and activities performed. The report is generated by the report generation application service.

Test Case 6 Security Audit Trail Log

The tester verifies that the security application service monitors the use of its own resources. The security audit trail will log the defined security events. The system will be able to create, maintain, and protect from modification, unauthorized access or destruction, an audit trail of accesses to the objects it protects.

Input: Security audit trail log, audit trail is active while processing of above test cases is occurring.

Output: The audit trail log will always contain the most current history of events. including the access and manipulation to secured element data by users/processes/elements and the date and time such accesses occurred. If a user deletes an object, for object deletion events, the audit record shall include the name of the object. The tester can use the previous test cases audit trail to verify this testes.

Expected Results: The function of continuous recording is available for recording security alerts and events.

Test Case 7 Verify the Administrators Ability to Delete a Directory Entry

The tester verifies that the administrator has the ability to delete/update entries in the directory.

Input: Several registered user profiles. Administrator logon and password. Access the directory. Delete a profile entry in the directory.

Output: The directory entry will be deleted. The registered user is no longer registered in the directory. If a user deletes an object, for object deletion events, the audit record shall include the name of the object. The tester can use the previous test cases audit trail to verify this testes.

Expected Results: The user is unable to modify an existing profile. The administrator access and activities are logged.

Test Case 8 User Disconnected

The tester verifies that ECS elements are relinquished after a user has logged onto the system and has not been active for a configurable period of time.

Input: User logs onto the system. The time period for disconnect is set to ten minutes. The user does not perform any activity on the system for a time greater than ten minutes.

Output: The user is disconnected after a period of ten minutes. The audit trail log will always contain the most current history of events. including the access and manipulation to secured element data by users/processes/elements and the date and time such accesses occurred. The time the user is disconnected from the ECS elements is recorded in the audit trail.

Expected Results: The time the user is disconnected from the ECS elements is recorded in the audit trail. The user must log into the system to regain access.

5.1.5.2.5 Sequence 5 - Preserving Data Integrity Across the Network

This sequence of test verifies the ability to preserve data integrity across a network, and the ability to copy data from one data facility to another. This test will verify that the data sent is the data received. Data messages are sent between two sites using Nascom or Ecom communication protocol. Data encryption and decryption devices are used to perform this task. The security of sending and receiving the data is made possible by the unique key that is agreed upon by the sender and receiver.

Test Case 1 Verify Data Transmitted is Data Received

The tester verifies that network logins are performed for valid user classes at EDF and GSFC. The two facilities will have remote capabilities of copying files.

Input: Request from EDF to copy a file down to GSFC. The user has access authority to copy the data file to the remote GSFC facility, DCE authorization and database authorization. The user will specify the desired node.

Output: User authorization and authentication are validated, and the file is transferred from EDF to GSFC. The data file transferred will be identical to the data file received.

Expected Results: The user will send a version of the file which will be placed in receiver's/target home directory. The data file transferred from EDF will be identical to the data file received at GSFC.

Test Case 2 Verify Recovery from System Failure

The tester verifies that data integrity is maintained in the event of a catastrophic failure.

Input: DAAC suffers a catastrophic failure.

Output: V0 data or other existing copy of the data is recovered. a snapshot of the system is taken at the point where viruses, security compromises and unscheduled system abort are detected. This information is used in restarting the system.

Expected Results: Irrespective of the data type i.e., data base entries or product, a version of the data exist and can be recovered at no cost to the user.

5.1.5.2.6 Sequence 6 - Security Risk and Compromise

This sequence of test verifies the ability to detect virus or worm penetration, and detection. Verification of virus monitoring and security compromises can be verified through report generation and verification. The sequence of test also verifies the CSMS operational concepts concerning system startup, shutdown, and maintaining data security and integrity.

Test Case 1 Verify Virus Monitoring

The tester verifies that the system is capable of monitoring for viruses prior to infecting the system to prevent system down time, and misuse of resources.

Input: Virus. Command to generate a report.

Output: An anti-virus check is performed periodically on storage media. When detected, the virus is cleaned from the file, and the file is stored with no compromises to data integrity. The administrator is able to generate a report detailing the detected viruses, and the remedy. All detected computer viruses are reported to the SMC.

Expected Results: The system is able to detect, clean, report viruses, and provide a remedy to the virus without compromising data integrity. The virus is reported to the SMC.

Test Case 2 Respond to Security Compromises

The tester verifies that the system is capable of responding to security compromises within a maximum of five minutes.

Input: A vital piece of information, password, which is part of the security of a file has been advertised unknowingly to the public.

Several guest users are able to access the file with the correct password.

Output: Within five minutes the system is able to detect a violation of the security of the file via the authority of the user. The user is notified. The event is logged and the security alert is generated at the security console. The event is also recorded in the audit trail.

Expected Results: The system is able to check the address, security level of a guest user, or discern the anonymity of a user. The integrity of the data is not compromised, and the event is logged.

Test Case 3 Verify System Abort and Recovery Procedures Maintain Data Integrity

The tester verifies that when the system aborts due to a catastrophic failure, it is able to recover gracefully. All data accesses are deleted, any data updates are recoverable, and data integrity is maintained.

Input: Simulated system abort, while the administrator is making updates to the directory database. Administrator issued warm start command.

Output: The directory database access privileges are deleted promoting object reuse, and uncompromised data. The administrator is notified. The event is logged and the security alert is generated at the security console. The event is also recorded in the audit trail. The administrator is able to restart the system and perform additional directory database cleanup. The old changes were stored when the system aborted and are recoverable. In the event the old changes were not saved, the most recent version of the dataset is recoverable.

Expected Results: The system is able to delete prior access privileges, when a system abort occurs. The integrity of the data is not compromised, and the event is logged.

Test Case 4 Verify Unscheduled Shutdown is Handled and Security Maintained

The tester verifies that the system is capable maintaining data integrity during an unscheduled shutdown. All data accesses are deleted, any data updates are recoverable, and data integrity is maintained.

Input: Issue command to shutdown a DAAC.

Output: All data and database access privileges are deleted promoting object reuse, and uncompromised security. The event is logged and the security alert is generated at the security console. The event is also recorded in the audit trail.

Expected Results: The system is able to delete prior access privileges, when a system shutdown occurs. The security of the data is not compromised, and the event is logged.

Test Case 5 Verify Scheduled Shutdown is Handled and Security Maintained

The tester verifies that the system is capable of a scheduled graceful termination, and that all data and database access privileges are deleted promoting object reuse, and uncompromised data.

Input: Issue command/automatic shutdown a DAAC.

Output: All data and database access privileges are deleted promoting object reuse, and uncompromised security. The event is logged and the security alert is generated at the security console. The event is also recorded in the audit trail.

Expected Results: The system is able to delete prior access privileges, when a system shutdown occurs. The security of the data is not compromised, and the event is logged.

5.1.6 System Integration Build 1 (B10)

The System Integration Build 1 is an integrated build of the following system threads:

- Enhanced System Ingest Thread
- Enhanced Algorithm Integration and Test Thread
- Product Management Thread
- Enhanced System Access Thread
- CSMS Security Thread

5.1.6.1 Build Objectives

The objective of the system integration build 1 is to regression test IR-1 threads to ensure that existing IR-1 capabilities have not regressed, support pre-processing of ancillary data, and test security software to ensure the effectiveness of built-in security controls. The objectives of the system integration build 1 are:

- Provide the capability to ingest ancillary data to support engineering and instrument data in the processing of science products.
- Provide the AIT tools to support algorithm integration and test.
- Provide product management to support the coordination of resource management and queue processing needed to generate a product.
- Provide system access capabilities over the network, for active and non-active principals.
- Provide the service to define the security parameters and domain for sensitive and general availability products.

5.1.6.2 Build Test Description

Testing is performed to verify the simultaneous processing of individual requests and to verify the acceptance of requests, allocation of resources to support data processing, resource authorization and authentication, and quality assurance of generated products. The algorithm ingest and resource utilization sequence verifies that science software packages can be ingested into the system, scheduled, and that resources can be allocated and managed in order to perform processing for individual requests for product generation.

The system security sequence verifies the ability to define an explicit set of security policies, set access controls, protect the quality and accessibility of data, and monitor user access. Virus detection, object reuse, and security compromise are also integral parts considered in testing system security.

Dependencies: (If Applicable)

In order to verify the objectives of the system integration build 1, the following interface and functional capabilities are required:

- Interface to SCF's to ingest science software delivery packages.
- Interface between SDPS resource management and CSMS.
- Interface to PGS toolkit and IMS toolkit to verify algorithm compliance and configuration management control.
- Interface to ECS via techniques such as Kerberized ftp and telnet.

The following customary service interfaces are required:

- E-mail Application interface.
- Scheduled file transfer interface.
- File transfer application interface.
- Bulletin board application interface.
- Event logging application interface.

The following object services are required:

- Directory/naming service.
- Time service.
- Security service.
- Event service.
- Message passing.

Test Support Requirements

- Hardware:
  - DCE workstation (HP735, SUN Sparc 10, DEC 3000/300, IBM RS6000, SGI R3000, and CRAY)

- Software:
  –
- Data:
  – simulated data granules
  – metadata, algorithm executables, ancillary data products

5.1.6.2.1 Sequence 1 - System Ingest and Algorithm Integration with Resource Utilization

The following series of tests verifies that science software delivery packages, L0 data, and associated ancillary data can be ingested into the system. Data is ingested. If errors are encountered during ingest, a retransmission request is generated. If no errors are encountered data validation is performed. Testing verifies that the proper resources are allocated to support data ingest, staging, verification, pre-processing (if required), and storage. Resources are allocated to support staging of an algorithm package, verification of algorithm compliance and storage.

Test Case 1 Successful Science Software Delivery Package

This test verifies that a Science Software Delivery Package can be transferred from a SCF to a representative DAAC host in the EDF. The Science Software Delivery Package must contain all the supporting documentation for the associated algorithm (test files, processing results, etc). This test verifies that each required part of the package is transferred successfully. Resources are allocated to support the transfer of the algorithm package

Input: Ingest of Science Software Delivery Package (algorithm). Interface to PGS toolkit to verify algorithm compliance. Interface to resource allocation to support allocation of resources to support algorithm ingest, staging, compliance verification, preliminary testing of the algorithm, and storage. Data file to support testing of algorithm processing functionality.

Output: Resources are properly allocated to support ingesting of the entire Science Software Delivery Package, staging of the algorithm, processing of test data file to verify algorithm processing functionality, and storage. Algorithm is verified to be in compliance with defined standards, no errors encountered during test processing. Algorithm written to permanent storage. Directory updated. Ingest log contains entry for ingest initiation and completion. Resource log contains an entry for allocation of resources for ingest staging, PGS toolkit allocation, processing results, archival, and de-allocation of resources upon completion.

Expected Results: Algorithm ingested, verified, and written to permanent storage. Resources allocated to support ingest and staging properly relinquished upon algorithm permanent archive.

Test Case 2 Resource Allocation for Ancillary Data

This test verifies that resources are properly allocated to support ingesting and pre-processing of ancillary data. Resource allocation includes allocation for data staging, algorithm to support pre-processing, allocation for processing results, and data storage.

Input: Ingest of ancillary data requiring pre-processing. Algorithm to be used for pre-processing of ancillary data.

Output: Ancillary data ingested and validated. Algorithm scheduled for pre-processing. Pre-processing completed. Ancillary data written to permanent archive. Ingest log contains an entry for ingest initiation and completion. Resource log contains an entry for allocation of resources to support ingesting of ancillary data, allocation of algorithm used for pre-processing, staging of processing results, archival of data, and de-allocation of all resources upon completion.

Expected Results: Ancillary data is ingested and pre-processing performed. Resources are properly allocated and de-allocated as required. Data availability notice generated upon successful data archival.

Test Case 3 Resource Allocation for Ingesting Metadata

This test verifies the capability to ingest metadata. Testing verifies that resources are properly allocated for ingest data staging, data validation, and permanent archiving.

Input: Ingest of metadata. Resource allocation to support data ingest, staging, and permanent archiving.

Output: Resources properly allocated to support data ingest, staging, verification, and permanent archiving. Metadata successfully ingested, verified, and written to permanent archive. Resource log contains an entry for the allocation of resources for data staging, verification, and archival. Ingest log contains an entries for ingest initiation and completion. Data availability notice generated after successful archive.

Expected Results: Resources properly allocated and de-allocated as required. Data availability notice generated upon successful data archival.

Test Case 4 Resource Allocation for Ingesting V0 Data

This test verifies the capability to ingest V0 data Testing verifies that resources are properly allocated for ingest data staging and permanent archiving.

Input: Ingest of V0 data. Resource allocation to support data ingest, staging, and permanent archiving.

Output: Resources properly allocated to support data ingest, staging, and permanent archiving. V0 data successfully ingested and written to permanent archive. Resource log contains an entry for the allocation of resources for data staging and archival. Ingest log

contains an entries for ingest initiation and completion. Data availability notice generated after successful archive.

Expected Results: Resources properly allocated and de-allocated as required. Data availability notice generated upon successful data archival.

Test Case 5 Resource Allocation for Ingesting Data Requiring Conversion/ Reformatting

This test verifies the capability to ingest data requiring conversion/reformatting. Testing verifies that resources are properly allocated for ingest data staging, data conversion/reformatting, and permanent archiving.

Input: Ingest of data requiring conversion/reformatting. Resource allocation to support data ingest, staging, conversion/reformatting, and permanent archiving.

Output: Resources properly allocated to support data ingest, staging, conversion/reformatting and permanent archiving. Data successfully ingested, converted/reformatted and written to permanent archive. Resource log contains an entry for the allocation of resources for data staging, data conversion/reformatting and archival. Ingest log contains an entries for ingest initiation and completion. Data availability notice generated after successful archive.

Expected Results: Resources properly allocated and de-allocated as required. Data availability notice generated upon successful data archival.

Test Case 6 Scheduling an Algorithm for Execution

This test verifies the capability to schedule an algorithm for processing. Allocation of all required resources to support use of algorithm in processing production environment is verified.

Input: Algorithm scheduler, interface to resource management, algorithm to be scheduled.

Output: Allocation of all required resources to support algorithm processing in the production environment. Resource log contains entry for the allocation of the resources for algorithm processing. Event log contains entry for the scheduling of an algorithm using the algorithm scheduler.

Expected Results: Resources properly allocated to support algorithm processing in the production environment.

5.1.6.2.2 Sequence 2 - Enhanced System Access and CSMS Security

This series of tests verifies policies and procedures either written or implemented by control parameters provide the guidance necessary for sites to accomplish mission objectives. Enhanced system access and network security are governed by the policies and procedures of the

procedural process which supports IR-1 access to both local and remote systems/networks and data security, and the use of office automation tools.

Using the policies and procedures of accessing the system, testing will verify the ability to access site resources through authentication and authorization of users in a client/server environment. The system will prevent intrusion, relinquish connection, while maintaining data integrity, object reuse, and privacy. Recovery procedures will be verified. The security procedures will adhere to C2 level security guidelines, and will be established through coordination with site managers. Testing will verify the authorized uses of system resources by allowing the administrator to define the principals or domain, and the permissions in the access control list (ACL).

Test Case 1 — Control Parameters for ECS Data Access on the LAN using E-mail

This test verifies that an authorized user can access the system after the policies and procedures defined using control parameters are set. User classes and permissions are parameters which when set and administered determine the resources and users allowed to access the system and data.

Input: Define user classes -*guest* and *registered.*

Define access controls for public data as *read* and assign the permission to the guest user class.

Define access controls for non-public data as *read* and assign the permission to the registered user class.

Define the security events.

The tester performs a local logon - rlogin into the system and request user registration from a local host by sending an e-mail message across the LAN.

Output: The user is able to securely logon to the system and generate a e-mail message requesting registration application for ECS services. The message is received at the receiving address, and the recipient is able to access and read the message. The registration application is returned and processed. The user is now a registered user.

Expected Results: The user's registration status is registered, and has access authority as defined by the Registered class.

Test Case 2 — Control Parameters for ECS Data Access on the WAN using BBs Link

This test verifies that an authorized user can access the system after the policies and procedures defined using control parameters are set. User classes and permissions are parameters which when set and administered determine the resources and users allowed to access the system and data.

Input:	Define user classes -*guest and registered.*
	Define access controls for public data as *read* and assign the permission to the guest user class.
	Define access controls for non-public data as *read* and assign the permission to the registered user class.
	Define the security events.
	The tester performs a remote logon - Telnet into the system and request user registration from another host on a WAN by sending an message using BBs link across the WAN.
Output:	The user is able to securely logon to the system and generate a message requesting registration application for ECS services. The message is received at the receiving address, and the recipient is able to access and read the message. The registration application is returned and processed. The user is now a registered and access and activities are logged.
Expected Results:	The user's registration status is registered, and has access authority as defined by the Registered class.

Test Case 3 User Registration - Existing Profile

The tester verifies the ability of a registered user to access the directory and modify an existing profile and obtain higher privileges than a user without an existing profile. Successful downloading of the scientist workbench to the local register's workstation is required.

Input:	Registered user login, obtain and complete the on-line registration application request. The tester should define and modify the following user profile information to include: data distribution address and media, default query parameters, user electronic address, user expertise level, terminal characteristics, and technical specialty. All other information is protected by access control.
Output:	ECS service application received and processed. Application is processed . User now has higher privileges to the ECS system.
Expected Results:	A higher user access authority is granted. The user can test access authority by requesting data. User access and activities are logged.

Test Case 4 Verify Compliance to Security Directives across Network

The tester verifies the security events, and access controls are complied to when accessing and sending data across the network via the file transfer service. The tester verifies that network logins are performed for a valid user class at EDF and GSFC. The two facilities have remote capabilities of copying files. After the file transfer, verify that the user is not actively using the system for a configurable period of time, and is disconnected from the system.

Input: The security access controls are set, and the security events are defined. A request is made from EDF to perform a WAN transfer of a data file less than 5 Meg to GSFC. The user has access authority to copy the data file to the remote GSFC facility, DCE authorization and database authorization. The user specifies the desired node. The time period set for disconnect from the system for an inactive address is ten minutes. After requesting the file transfer, do not use the system for a time period less than ten minutes.

Output: User authorization and authentication are validated, and the file is transferred from EDF to GSFC. The event log contains the entry for the login and the transfer of the data file. The user is disconnected after a period of ten minutes. The time the user is disconnected from the ECS element is recorded in the audit trail.

Expected Results: The user will send a version of the file which will be placed in receiver's/target home directory. The data file transferred will be identical to the data file received at GSFC. The user is disconnected from the system.

Test Case 5 Object Reuse

The tester verifies the ability of the system to revoke all authorizations to information prior to assignment, allocation, or reallocation to another user. This test will verify that a prior users, user disconnected in testcase 4, access to an object has been released back to the system, and no subsequent user will inherit prior access privileges.

Input: Repeat testcase 4 and transfer non-public data. A guest user attempts to perform the steps in testcase 4 for which the guest user does not have access to the non-public data.

Output: The guest user is not able to transfer the non-public data.

Expected Results: Access authority is granted to the registered user, and is not granted to the guest user. User access and activities are logged.

Test Case 6 Posting and Retrieving Articles on the Bulletin Board

This test verifies that an authorized user can post and retrieve articles to and from the bulletin board.

Input: Article item to post to the bulletin board.

Article item to retrieve from the bulletin board.

Output: Through the bulletin board link, the user is able to post and retrieve articles on the bulletin board.

Expected Results: The event is completed successfully an logged.

Test Case 7 Unauthorized Access to ECS System

This test verifies that an unauthorized user cannot log onto the ECS system and use it such as posting articles to the bulletin board. The tester attempts to log on with an invalid logon id. The system logs the user on as a guest and asks the tester to register. The user attempts to by pass registration to gain entry into the system. Access is denied. The event is logged.

Input: Unauthorized user id and password.

Output: The user is denied system access.

Expected Results: The invalid user is requested to register into the system and is denied entry.

Test Case 8 Verify Virus Monitoring

The tester verifies that the system is capable of monitoring for viruses prior to infecting the system to prevent system down time, and misuse of resources.

Input: Virus. Command to generate a report.

Output: An anti-virus check is performed periodically on storage media. When detected, the virus is cleaned from the file, and the file is stored with no compromises to data integrity. The administrator is able to generate a report detailing the detected viruses, and the remedy. All detected computer viruses are reported to the SMC.

Expected Results: The system is able to detect, clean, report viruses, and provide a remedy to the virus without compromising data integrity. The virus is reported to the SMC.

Test Case 9 Respond to Security Compromises

The tester verifies that the system is capable of responding to security compromises within a maximum of five minutes.

Input: A vital piece of information, password, which is part of the security of a file has been advertised unknowingly to the public. Several guest users are able to access the file with the correct password.

Output: Within five minutes the system is able to detect a violation of the security of the file via the authority of the user. The user is notified. The event is logged and the security alert is generated at the security console. The event is also recorded in the audit trail.

Expected Results: The system is able to check the address, security level of a guest user, or discern the anonymity of a user. The integrity of the data is not compromised, and the event is logged.

Test Case 10 Security Audit Trail Log

The tester verifies that the security application service monitors the use of its own resources. The security audit trail, a component of network security, will monitor the defined security events. The system will be able to create, maintain, and protect from modification or unauthorized access or destruction, an audit trail of accesses to the objects it protects.

Input:	Security audit trail log, audit trail is active while processing of above test cases is occurring.
Output:	The audit trail log will always contain the most current history of events, including the access and manipulation to secured element data by users/processes/elements and the date and time such accesses occurred. If a user deletes an object, for object deletion events, the audit record shall include the name of the object. The tester can use the previous test cases audit trail to verify this test.
Expected Results:	The function of continuous recording is available for recording security alerts and events.

5.1.7 Client Services Thread (T11-01)

5.1.7.1 Thread Objectives

The objectives of the client services thread are as follows:

- Provide the capability to perform a user registration and obtain the ECS desktop and workbench services.
- Provide a 'Help' function using a GUI interface with useful/meaningful information being supplied to the user.
- Provide data visualization capabilities (EOSView) to support viewing of HDF data.
- Provide a user environment (via Scientist workbench) that supports EOSDIS data services.

5.1.7.2 Thread Test Description

Testing performed under this thread will verify that different privileges are established during user registration based on the existence or non-existence of profile information for the individual completing the registration. Registration will include the successful downloading of the scientist workbench to the register's local workstation. Options available on the workbench will vary depending on the privileges granted during the registration.

Testing to verify the client can access the GUI 'help' function will be demonstrated. Help data will be read to determine if the information contained will enable the 'novice' user to navigate through the ECS system with little difficulty.

Testing to verify the functionality of EOSView will be performed. EOSView will be accessible to the user via the scientist workbench. Testing will demonstrate the capability to view HDF-EOS data files.

Testing of the scientist workbench to verify the various GUI and CHUI interfaces to the EOSDIS and ECS services will be performed. The tester demonstrate the capability to customize the look and feel of the scientist workbench to support the viewing of and managing of data manipulation based on user preference.

Dependencies: (If Applicable)

In order to verify the objectives of the client services thread, the following interface and functional capabilities are required:

- Network environment (LAN and WAN).
- User registration capability, scientist workbench, and EOSView.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  -
- Data:
  - HDF data files.

5.1.7.2.1 Sequence 1 - User Registration

The following series of tests verify that different privileges are established during user registration based on the existence or non-existence of profile information for the individual completing the registration. Registration will include the successful downloading of the scientist workbench to the register's local workstation. Options available on the workbench will vary depending on the privileges granted during the registration.

<u>Test Case 1</u> <u>User Registration - Existing Profile Information</u>

This test verifies that a user completing registration that has existing profile information is granted higher privileges than a user without existing profile information. This test case is executed simulating a SCF user registration, DAAC user registration, ADC registration, and other external user registration. Successful downloading of the scientist workbench to the local register's workstation is required.

Input: Log onto ECS system. Fill out the on-line registration form. The registration information includes name, address, organization affiliation, requested services, e-mail address, network, etc.

Output: User registration complete. User granted privileges to access the ECS system. Successful download of scientist workbench to the local workstation successful.

Expected Results: User able to bring up the scientist workbench and obtain access to the ECS system.

Test Case 2 User Registration - No Profile Information

This test verifies that a user completing registration that does not have existing profile information is not granted the same privileges as a user with existing profile information. This test case is executed simulating a SCF user registration, DAAC user registration, ADC registration, and other external user registration. Successful downloading of the scientist workbench to the local register's workstation is required.

Input: Log onto ECS system. Fill out the on-line registration form. The registration information includes name, address, organization affiliation, requested services, e-mail address, network, etc.

Output: User registration complete. User granted privileges to access the ECS system. Successful download of scientist workbench to the local workstation successful.

Expected Results: User able to bring up the scientist workbench and obtain access to the ECS system.

Test Case 3 Comparing Results of Test Case 1 and Test Case 2

This test verifies that the user registrations performed in Test Case 1 and Test Case 2 resulted in different privileges being granted to the user based on the existence or non-existence of profile information for that register.

Input: Comparison of results of registration details from executing Test Case 1 and Test Case 2.

Output: Differences of privileges granted to users completing registration information as outlined in Test Case 1 and Test Case 2.

Expected Results: Privileges granted during execution of Test Case 1 should be higher than those granted during execution of Test Case 2.

5.1.7.2.2 Sequence 2 - Invalid Registration

The following test verifies that error messages are provided to aid the user during the registration process.

Test Case 1 Invalid Field Entry in the Registration Form

This test verifies that error messages are generated during the pre-registration process as a result of entering invalid registration criteria in the registration form.

Input: Fill out the on-line registration form with invalid registration information.

Output: The user receives an error message each time invalid information is entered into each field on the registration form and is provided with information on the proper format/data that should be entered into the field.

Expected Results: The user can not continue to fill out the registration form until the incorrect field has been corrected. Invalid forms can not be submitted for processing.

5.1.7.2.3 Sequence 3 - Utilizing Help Functions

The following tests verify the use of the menu driven and context sensitive help functions.

Test Case 1 Menu Driven Help

This test verifies the capability of a menu driven help function to find general information. The help option is accessible to the user via the scientist workbench.

Input: Menu driven help screen. Operator selectable help options. Operator selects option for additional help information.

Output: Help information is provided about the option selected. Information provided will aid any level of user (novice to well experienced).

Expected Results: The help information is displayed to the screen. Operator has an option to print the help information if desired.

Test Case 2 Context Sensitive Help

This test verifies the ability to access context sensitive help functions. Context sensitive help enables the user to obtain definitions for words that are unfamiliar by selecting (highlighting) the word and clicking the 'help' option. Tester will access context sensitive help from the scientist workbench.

Input: Highlight keywords for searching. Click on the highlighted word to obtain specific context information about the keyword.

Output: Meaningful definition/explanation about the item is displayed.

Expected Results: The tester is given detailed information about the particular item that was highlighted.

5.1.7.2.4 Sequence 4 - Data Visualization Capabilities Using EOSView

The following series of tests verify the capability to view HDF-EOS data files using EOSView. EOSView is installed onto the scientist workbench from a list of available tools using the 'drag and drop' capability. Testing is performed to verify that the user can view HDF data files using panning, zooming, palette selection, and animation.

Test Case 1 Verify EOSView API Functionality

This test verifies the client is able to load the EOSView application onto the local scientist workbench and that successive activation of the application is successful.

Input: Scientist workbench. List of tools available to the user. User installs the EOSView application to the scientist workbench using the 'drag and drop' option.

Output: Icon of EOSView application installed on the users scientist workbench. Activation of the application results in EOSView main menu presentation. Event log contains entries for installation of EOSView application to the user's local workbench and the successive activation of the application.

Expected Results: User able to install EOSView application on the workbench and successfully activate the tool after installation.

Test Case 2 Image Display and Manipulation

This test verifies the functionality of the image manipulation capabilities available in EOSView. Images can be manipulated by palette selection, panning, and zooming. The user selects an available HDF data file (i.e., raster image group) The image is loaded and displayed. This test case is executed for each data group available.

Input: Selection of HDF data group from EOSView (i.e., raster image group). Options available for manipulation of the data displayed are exercised (i.e., panning, zooming).

Output: The selected image is loaded and displayed. The view is updated with each option of data viewing selected. Event log contains an entry for the loading of the HDF data selected.

Expected Results: The selected data is loaded and displayed. The user is able to exercise the various manipulation capabilities on the loaded data.

Test Case 3 Animation

This test verifies the functionality of the image animation capabilities available in EOSView. Animations can be manipulated by controlling the speed of the image, going backward and forward one image at time or multiple images at time, and using the "bounce" mode to bounce from image(s) to image(s). This test case is executed for a single image and multiple images.

Input: Selection of HDF data group from EOSView (i.e., raster image(s)). Exercise the animation capabilities for the image(s) displayed.

Output: Selected image(s) is loaded and displayed. The view is updated with each animation selection. Event log contains an entry for the loading of the HDF data selected.

Expected Results: The selected image(s) is loaded and displayed. The user is able to exercise the various animation capabilities on the selected image(s).

Test Case 4 Multi-Dimensional Plots/Diagrams

This test verifies the functionality of generating and displaying multi-dimensional plots/diagrams (i.e., x-y plots, scatter plots, profiles, histograms, and surface diagrams) using capabilities available in EOSView.

Input: HDF data file. Select option to display data in multi-dimensional plots/diagrams.

Output: The HDF data file is loaded and displayed in multi-dimensional plots/diagrams. Event log contains an entry for the loading of the HDF data file selected.

Expected Results: User is able to view HDF data in multi-dimension plots/diagrams.

Test Case 5 Contour Plots

This test verifies the functionality of generating and displaying contour plots using capabilities available in EOSView.

Input: HDF data file. Select option to view the data in contour plots.

Output: The HDF data file in contour plots. Event log contains an entry for the loading of the HDF data file selected.

Expected Results: User is able to view HDF data file in contour plots.

5.1.7.2.5 Sequence 5 - Toolkit Interfaces

The following series of tests verify the capability to access the toolkits on the scientist workbench. The initial version of the scientist workbench downloaded to the register's local workstation has some toolkits installed as a default. Verification of the accessibility of these toolkits will be performed. Additional testing will be performed to verify that the user can install other ECS toolkits onto the local scientist workbench to customize the look and feel of the user's workbench.

Test Case 1 Verifying Toolkit Accessibility

This test verifies the accessibility of the toolkits delivered as part of the scientist workbench. This test case is executed for each toolkit provided as a part of the initial scientist workbench configuration.

Input: Icon's on the scientist workbench that provide the interface between the workbench and the supported toolkit. Select icon.

Output: The toolkit selected is presented to the user. User is able to enter data into the menu driven toolkit interface and receive valid data feedback.

Expected Results: Toolkit is accessible to the user. User is presented with a menu driven interface to the toolkit under test.

Test Case 2 Installing New Toolkits

This test verifies the capability of the user to install additional toolkits available to the ECS system on the users local scientist workbench.

Input: User is provided with a list of available toolkits for use on the scientist workbench that were not initially provided with the workbench. User selects new toolkit to put on the local scientist workstation.

Output: The new toolkit is successfully loaded onto the local scientist workbench. An icon is displayed on the workbench for the toolkit.

Expected Results: User is able to click on the icon and the toolkit menu interface is presented.

Test Case 3 Customizing the Desktop

This test verifies the capability of the user to customize the desktop, changing default settings and user preferences.

Input: User desktop for the ECS system. User customizes the look and feel of the desktop to support individual needs (i.e., change menu colors, remove icons to functions that will not be used, install additional services).

Output: The desktop is updated with changes specified by the user. Un-wanted icon's are removed from the desktop, new services are installed and displayed as icons on the desktop, and user is able to change the color of desktop displays/menu items. History log contains entries for any changes made to the desktop.

Expected Results: User is able to customize the desktop to support user needs.

Test Case 4 Supporting DAAC Data Visualization Utilities

This test verifies the capability to interface with DAAC specific data visualization utilities using the scientist workbench.

Input: ECS scientist workbench, DAAC specific data visualization utilities.

Output: User is able to access and utilize DAAC specific data visualization utilities from the scientist workbench. User does not have to exist workbench in order to utilize the utilities.

Expected Results: User is able to activate the utility from the workbench. Utility does not terminate abnormally after activation. Data visualization options function normally.

5.1.7.2.6 Sequence 6 - User Access to Libraries

The following test verifies that users are able to obtain access to the system libraries.

Test Case 1 Accessing the Libraries

This test verifies user access to the system libraries.

Input:	System libraries. Valid user logon. Transfer desired library to local workstation.
Output:	Review of library desired by user (i.e., mathematical library). Event log contains an entry for user logon and retrieval of library.
Expected Results:	User able to access system library and transfer desired library to local workstation.

5.1.8 Search Services Thread (T11-02)

5.1.8.1 Thread Objectives

The objective of the search services thread is to demonstrate the ability to perform a search of the data server guide, directory, and inventory tables as well as performing a search for documentation. Searching based on keywords is supported.

Search results are provided in a user specified sorted criteria, coverage maps, view details, and a results list.

The objectives of this thread are to provide the following search functions:

- Graphical data search to include directory, guide, and inventory.
- Character based inventory and guide searches.
- Spatial metadata search to include global granules, points/radius, and bounding box tools.
- Core metadata searching to include graphical and character based searches.
- Time coincidence assessment searching.
- Start and stop time searching.
- Building a complex query of search criteria.
- Dependent valids method of building search criteria.
- Searching for CERES, and TRMM data using keywords.
- Sorting results package by keywords for CERES, and TRMM data.
- Coincidence data viewing.

5.1.8.2 Thread Test Description

This thread test covers all the means by which a user may prepare a search request for data and verify the results returned from a search request. The tester builds search requests using each of the defined methods and examines the returned result package for validity and completeness.

Dependencies: (If Applicable)

In order to verify the objectives of the search services thread, the following interface and functional capabilities are required:

- Fully loaded data server with CERES and TRMM simulated data to include metadata, land and ocean inventory data, directory data, and guide data.
- Functional data dictionary to use during search request generation.
- Functional LIM to respond to search requests.
- Access to the methods used to build search criteria to include map, time line, and lat/long tools.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  - Graphical Map Search Selection Tool
  - Time Line Search Selection Tool
  - Graphical Lat/Long Search Selection Tool
- Data:
  - CERES, and TRMM data
  - Guide data
  - Land and Ocean inventory data
  - Metadata

5.1.8.2.1 Sequence 1 - Graphical Directory Search

The following series of tests verify a user can perform a graphical directory search. A graphical directory search can utilize a LIM or a data server based on the context. The test cases which follow access data from a LIM and a data server respectively. In order to perform these test cases, the tester must have knowledge of data sets which reside in a unique LIM and a unique data server in order to validate the test results.

<u>Test Case 1</u> <u>LIM graphical directory search</u>

This test verifies that a user can perform a graphical directory search of information controlled by a LIM. The tester selects the graphical region which contains the LIM controlled data and submits the search request. The result package is represented graphically.

Input: Data sets controlled by the LIM, search request.

Output: Search result package.

Expected Results: The returned search result package matches the information previously obtained by the tester prior to test execution. A comparison is performed to verify the LIM search was successful.

Test Case 2 Data Server graphical directory search

This test verifies that a user can perform a graphical directory search of information controlled by a data server. The tester selects the graphical region which contains the data server controlled data and submits the search request. The result package is represented graphically.

Input: Data sets controlled by the data server, search request.

Output: Search result package.

Expected Results: The returned search result package matches the information previously collected by the tester prior to test execution. A comparison is performed to verify that the data server search was successful.

Test Case 3 Non-existent graphical data search request

This test verifies that the search results package contains the appropriate messages for graphical directory search of regions containing no data. The tester selects a region on a map for which no data is available. The search result package reflects that no data was found.

Input: Graphical region containing no data, search request.

Output: Search result package.

Expected Results: The search result package reflecting no data was found.

5.1.8.2.2 Sequence 2 - Guide Search

The following series of tests verify a user can perform graphical and textual guide searches. A graphical guide search searches a DAAC for text type information. A textual guide search accepts text strings and returns related guide information.

Test Case 1 Build search string

This test verifies that a user can utilize the on-line help mechanisms while formulating a search string against a DAAC guide. The tester demonstrates the various help mechanisms (i.e., the data dictionary) in creating the search string. Once the string is completed, the search is initiated and the results of the guide search returned.

Input: Graphical guide search request.

Output: Search result package.

Expected Results: The returned search result package contains textual information which matches the information in the search string.

Test Case 2 Text search string

This test verifies that a user can input textual information to formulate a search string against a DAAC guide. The tester demonstrates how the system responds to input of meaningful textual strings. An example of this type of search is to input the string 'DAAC'. Submitting this string results in all documents concerning 'DAAC' issues being returned. Another example is to input the string 'snow and ice'. When this string is submitted, all documentation related to 'snow and ice' studies is returned.

Input: Textual guide search request.

Output: Search result package.

Expected Results: The returned search result package contains textual information which matches the information in the search string.

Test Case 3 Non-existent graphical guide data search request

This test verifies that the search results package contains the appropriate messages for a graphical guide search containing no data. The tester must obtain knowledge of a region for which no guide data exists. The tester then submits the request and verifies the returned results package.

Input: Search string for which there exits no data.

Output: Search result package.

Expected Results: The search result package reflecting no data was found.

Test Case 4 Non-existent textual guide search request

This test verifies that the search results package contains the appropriate messages for a textual directory search containing no data. The tester selects a text string unrelated to any documents in the system. The search result package reflects that no data was found.

Input: Textual search string unrelated to any guide data.

Output: Search result package.

Expected Results: The search result package reflecting no data was found.

5.1.8.2.3 Sequence 3 - Graphical Inventory Search

This sequence of tests verifies that land and ocean data inventory can be searched using graphical tools. The tester utilizes the mapping tool and the time line tool to construct the search requests.

Test Case 1 Graphical Land Data Inventory Search

The tester constructs a global request for land data using the mapping tool provided for this thread. After the global region is selected, the tester uses the graphical time line tool to select a time period for the global graphical data. The tester then submits the request and verifies the returned search result package.

Input: Search request, Valid global region, Valid time period.

Output: Search result package.

Expected Results: The search result package contains all the data in the global region selected for the time period specified.

Test Case 2 Graphical Land Data Region Inventory Search with no data found

The tester constructs a global request for land data using the mapping tool provided for this thread. The region selected is one for which no data exists. The tester then submits the request and verifies the returned search result package.

Input: Search request, Global region which contains no data.

Output: Search result package.

Expected Results: The search result package stating that no data was found for either of the regions selected.

Test Case 3 Graphical Land Data Time Line Inventory Search with no data found

The tester constructs a time line request for land data. The time period selected produces a search result with no data found. The tester then submits the request and verifies the returned search result package.

Input: Search request, Time period for which no data exists.

Output: Search result package.

Expected Results: The search result package states that no data was found for the time period selected.

Test Case 4 Graphical Ocean Data Inventory Search

The tester constructs a global request for ocean data using the mapping tool. After the global region is selected, the tester uses the graphical lat/long line tool to select a more defined area for the graphical search. The tester then submits the request and verifies the returned search result package.

Input: Search request, Valid ocean region containing data.

Output: Search result package.

Expected Results: The search result package contains all ocean data in the region selected for the lat/long positions specified.

Test Case 5 Graphical Ocean Data Region Inventory Search with no data found

The tester constructs a global request for ocean data using the mapping tool provided for this thread. The region selected is one for which no data exists. The tester then submits the request and verifies the returned search result package.

Input: Search request, Global region which contains no data.

Output: Search result package.

Expected Results: The search result package states that no data was found for the region selected.

Test Case 6 Graphical Ocean Data Lat/Long Tool Inventory Search with no data found

The tester constructs a request for ocean data using the graphical lat/long tool. The area selected contains no ocean data. The tester then submits the request and verifies the returned search result package.

Input: Search request, Time period for which no data exists.

Output: Search result package.

Expected Results: The search result package states that no data was found for either of the regions selected.

5.1.8.2.4 Sequence 4 - Character Inventory Search

This sequence of tests verifies that land and ocean data inventory can be searched using character specifications. The tester chooses options from each character category represented in the search window.

Test Case 1 Character Land Data Inventory Search

The tester constructs a request for land data using the various character search specifications available. The tester chooses options from each character category represented in the build search window. The tester then submits the request and verifies the returned search result package.

Input: Search request, Valid character fields.

Output: Search result package.

Expected Results: The search result package contains all data matching the search criteria specified in the search request.

Test Case 2 Character Ocean Data Inventory Search

The tester constructs a request for ocean data using the various character search specifications available. The tester chooses options from each character category represented in the build search window. The tester then submits the request and verifies the returned search result package.

Input: Search request, Valid character fields.

Output: Search result package.

Expected Results: The search result package contains all ocean data matching the search criteria specified in the search request.

Test Case 3 Character Land Data Inventory Search with no data found

The tester constructs a request for land data using the various character search specifications available. The options selected from each character category represented in the build search window results in no data being found. The tester then submits the request and verifies the returned search result package.

Input: Search request, Character fields which results in no data being found.

Output: Search result package.

Expected Results: The search result package reflects that no data was found.

Test Case 4 Character Ocean Data Inventory Search with no data found

The tester constructs a request for ocean data using the various character search specifications available. The options selected from each character category represented in the build search window results in no data being found. The tester then submits the request and verifies the returned search result package.

Input: Search request, Character fields which results in no data being found.

Output: Search result package.

Expected Results: The search result package reflects that no data was found.

5.1.8.2.5 Sequence 5 - Spatial Metadata Search

The following series of tests verify a user can perform a spatial metadata graphical inventory search. A spatial metadata search can be expressed as global granules, points/radius, or bounding box using a map. All of these methods are tested in this sequence.

Test Case 1 Global Granule Test 1

This test verifies that a user can perform a graphical inventory search using global land attributes. The tester uses the mapping tool to bring up a global view of the earth. The tester then selects to perform a global land search on a selected data type. The search result package is verified against search criteria.

Input: Mapping tool, global land request.

Output: All data available for the selected data type.

Expected Results: The search result package contains all the land data for the selected data type.

Test Case 2 Global Granule Test 2

This test verifies that a user can perform a graphical inventory search using global ocean attributes. The tester uses the mapping tool to bring up a global view of the earth. The tester then selects to perform a global ocean search on a selected data type. The search result package is verified against the search criteria.

Input: Mapping tool, global ocean request.

Output: All data available for the selected data type.

Expected Results: The search result package contains all the ocean data for the selected data type.

Test Case 3 Point/Radius Search Test A - Draw

This test verifies that a user can select a circular area of interest on a map and submit a search request on the area contained by the circle. The tester uses the cursor to draw a circle on a map and submit a search request. The result package is verified to insure the returned data falls within the circular boundaries. The result package is represented graphically.

Input: Circle, search request.

Output: Search result package.

Expected Results: The returned search result package contains data which is contained within the circular boundaries.

Test Case 4 Point/Radius Search Test B - Select Points

This test verifies that a user can select a circular area of interest on a map and submit a search request on the area contained by the circle. The tester inputs the center point of the circle and the radius. The units for the circle can be selected as well as the geographical reference (lat./long. or UTM). The circle is drawn on the map and the tester can submit the request. The result package is verified to insure the returned data falls within the circular boundaries. The result package is represented graphically.

Input: Center point, radius length, geographical reference, search request.

Output: Search result package.

Expected Results: The returned search result package contains data which is contained within the circular boundaries.

Test Case 5 Point/Radius Search Test C - No data

This test verifies that when a user selects a circular area of interest on a map (for which no data exists) and submits a search request on the area contained by the circle, that appropriate messages are returned in the search result package. The tester inputs the center point of the circle and the radius for a area which no data exits. The circle is drawn on the map and the tester can submit the request. The result package is verified to insure that appropriate messages are returned. The result package is represented graphically.

Input: Center point, radius length, geographical reference, search request.

Output: Search result package.

Expected Results: The returned search result package contains appropriate messages indicating that no data was found.

Test Case 6 Bounding Box Search Test A - Draw

This test verifies that a user can select a rectangular area of interest on a map and submit a search request on the area contained by the rectangle. The tester uses the cursor to draw a rectangle on a map and submit a search request. The result package is verified to insure the returned data falls within the rectangular boundaries. The result package is represented graphically.

Input: Bounding box, search request.

Output: Search result package.

Expected Results: The returned search result package contains data which is contained within the rectangular boundaries.

Test Case 7 Bounding Box Search Test B - Select corner points

This test verifies that a user can select a rectangular area of interest on a map and submit a search request on the area contained by the rectangle. The tester inputs two latitudinal and two longitudinal points of interest. The bounding box is then drawn on the map and the tester then submits the search request. The result package is verified to insure the returned data falls within the rectangular boundaries. The result package is represented graphically.

Input: Bounding box latitudinal and longitudinal points, search request.

Output: Search result package.

Expected Results: The returned search result package contains data which is contained within the rectangular boundaries.

Test Case 8 Bounding Box Search Test C - No data

This test verifies that if a user selects a rectangular area of interest on a map (for which no data exists) and submits a search request on the area contained by the rectangle, that the appropriate messages are returned in the search result package. The tester inputs two latitudinal and two longitudinal points of interest. The bounding box is then drawn on the map and the tester submits the search request. The result package is verified to insure that the no data found messages are returned.

Input: Bounding box latitudinal and longitudinal points, search request.

Output: Search result package.

Expected Results: The returned search result package contains appropriate messages indicating that no data was found.

5.1.8.2.6 Sequence 6 - Core Metadata Test (Graphical)

This sequence of tests verifies that search criteria can be constructed and executed from the list of core metadata attributes. Over 200 core metadata attributes have been defined. It is assumed that unit testing has verified that a search can be executed on each core metadata attribute. This

sequence of tests verifies that combinations of graphical oriented attributes can be constructed and executed. The result package is graphically represented.

Test Case 1 Time Search

The tester graphically selects a period of time using the time line tool and selects a core metadata search. The result package reflects any metadata found within the time period selected.

Input: Data period with metadata available.

Output: Search result package.

Expected Results: The search result package contains a graphical representation of the metadata found within the specified time.

Test Case 2 Latitude/Longitude Search

The tester graphically selects a region on the map using the mapping tool and selects a core metadata search. The result package reflects any metadata found within the selected region.

Input: Region with metadata available.

Output: Search result package.

Expected Results: The search result package contains a graphical representation of the metadata found within the specified region.

5.1.8.2.7 Sequence 7 - Core Metadata Test (Character)

This sequence of tests verifies that search criteria can be constructed and executed from the list of core metadata attributes. Over 200 core metadata attributes have been defined. It is assumed that unit testing has verified that a search can be executed on each core metadata attribute. This sequence of tests verifies that combinations of character based attributes can be constructed and executed. The result package contains character data matching the selected attributes.

Test Case 1 Algorithm Search

The tester constructs a textual search entering data describing algorithm packages to be searched for. Any combinations of the listed inputs can be used. The search result package contains all algorithm packages which matches the selected criteria.

Input: Valid Algorithm information to include: Algorithm Package Acceptance Date; Algorithm Package Description, Algorithm Package Intended Operating System; Algorithm Package Maturity Code, Algorithm Package Name, Algorithm Package Version, and Algorithm Test Data Description.

Output: Search result package.

Expected Results: The search result package contains all algorithms found which match the search criteria provided.

Test Case 2 Author/Date/Instrument Search

The tester constructs a textual search command entering data related to a documentation search. This command searches the core metadata available to match the author, date, and instrument parameters entered. The search result package matches the selected criteria.

Input: Valid combination of search criteria to include author, dates, and instrument parameters to include: Author, Author of ATBD, Author of Design Document, Author of Diagram Document, Author of Proc Document, Date of Reference Paper, Date of ATBD, Date of Design Document, Date of Diagram Document, Date of Proc Document, and Instrument Short Name.

Output: Search result package.

Expected Results: The search result package contains information found based on the search criteria.

Test Case 3 Horizontal/Vertical Search

The tester constructs a textual search entering horizontal and vertical search criteria. This search string searches the core metadata available to match the search criteria. The search result package matches the selected criteria.

Input: Valid combination of search criteria to include Horizontal Positional Accuracy Explanation, Horizontal Region Description Dataset, Horizontal Region Description Grouping, Horizontal Region Name, Horizontal Coverage Name Dataset, Horizontal Coverage Name Grouping, Horizontal Representation, Horizontal Spatial Coverage, Horizontal Spatial Format, Horizontal Spatial Resolution Representation, Horizontal Spatial Coverage Representation, Vertical Class, Vertical Coverage, Vertical Positional Accuracy, Vertical Region Description Dataset, Vertical Region Name, Vertical Spatial Coverage, Vertical Spatial Format, Vertical Spatial Resolution.

Output: Search Result Package.

Expected Results: The search result package contains a textual representation of data which matches the input search criteria.

5.1.8.2.8 Sequence 8- Time Coincidence Data Search

The following series of tests exercises the various methods that users can utilize time in creating data searches. Time can be represented in Year/Month/Day, Hour/Minute/Second, and Julian formats. Each format is tested.

Test Case 1 Valid Year/Month/Day Format Combinations

This test verifies data search request results from using the year/month/day format. The tester inputs various combinations of year/month/day and inspects the returned results package to ensure that only data within the year/month/day specifications are returned.

Input: Valid combinations of year/month/day format of the search request.

Output: Returned search results package.

Expected Results: The returned search results package is verified for content. The test also verifies that no extraneous data was provided.

Test Case 2 Invalid Year/Month/Day Format

This test verifies that a search request can not be constructed using invalid formats for the year/month/day search criteria. The tester attempts to enter values for month outside of the valid range (1-12) and values for day outside of the valid range (1-31). The tester also attempts to combine valid month entry with invalid day entry (such as Feb 30, September 31, etc.). The system responds with an invalid format alarm.

Input: Invalid combinations of year/month/day format for the search request.

Output: Invalid format alarms.

Expected Results: Alarm messages for invalid search criteria format is received for each attempt to enter an invalid format.

Test Case 3 Invalid Year/Month/Day Time Period

This test verifies that invalid time periods are detected. The tester attempts to enter a negative time period (stop time less than start time). Also, the tester enters a period for which their is no data (future time and time before data was collected). The system is able to respond to the invalid time periods prior to initiating a search request.

Input: Negative time periods, time periods outside of valid collection period.

Output: Alarms indicating invalid periods detected.

Expected Results: Alarm messages are generated for each attempt to input an invalid time period as well as an alarm stating the requested time period is outside of the data collection period.

Test Case 4 Valid Hour/Minute/Second Format Combinations

This test verifies data search request results from using the hour/minute/second format. The tester first enters the dates for the search. The tester then inputs various combinations of hour/minute/second and inspects the returned results package to ensure that only data within the hour/minute/second specification are returned.

Input: Valid combinations of hour/minute/second format of the search request.

Output: Returned search results package.

Expected Results: The returned search results package is verified for content.

Test Case 5 Invalid Hour/Minute/Second Format

This test verifies that a search request cannot be constructed using invalid formats for the hour/minute/second search criteria. The tester attempts to enter values for hours, minutes, and seconds outside of the valid range (1-24 for hours, 1-59 for minutes and seconds). The system responds with an invalid format alarm.

Input: Invalid combinations of hour/minute/second format for the search request.

Output: Invalid format alarms.

Expected Results: Alarm messages for invalid search criteria format is received for each attempt to enter an invalid format.

Test Case 6 Invalid Hour/Minute/Second Period

This test verifies that invalid time periods are detected. The tester attempts to enter a negative time period (stop time less than start time). The system is able to respond to the invalid time periods prior to initiating a search request.

Input: Negative time periods.

Output: Alarms indicating invalid periods detected.

Expected Results: Alarm messages are generated for the attempt to input an invalid time period.

Test Case 7 Valid Search Using Julian Date

This test verifies data search request results from using the Julian date format. The tester inputs various combinations of Julian dates and inspects the returned results package to ensure that only data within the Julian specification are returned.

Input: Valid combinations of the Julian date format of the search request.

Output: Returned search results package.

Expected Results: The returned search results package is verified for content.

Test Case 8 Invalid Julian Date Format

This test verifies that a search request cannot be constructed using invalid formats for Julian date. The tester attempts to enter negative values and values greater than 366. The system responds with an invalid format alarm.

Input: Invalid combinations of Julian date for the search request.

Output: Invalid format alarms.

Expected Results: Alarm messages for invalid search criteria format is received for each attempt to enter an invalid format.

Test Case 9 Invalid Julian Format Time Period

This test verifies that invalid time periods are detected. The tester attempts to enter a negative time period (stop time less than start time). The system is able to respond to the invalid time periods prior to initiating a search request.

Input: Negative time periods.

Output: Alarms indicating invalid periods detected.

Expected Results: Alarm messages is generated for the attempt to input an invalid time period.

5.1.8.2.9 Sequence 9 - Start/Stop Time Sequence

The following series of tests demonstrates the ability to search for data based on a data set start and stop times and start/stop revolution times. Valid and invalid combinations are tested. The test also verifies that no extraneous data was provided.

Test Case 1 Valid Data Set Start/Stop Time Combinations

This test verifies data search request results from using the data set start/stop time format. The tester inputs various combinations of data set start and stop times and inspects the returned results package to ensure that only data within the data set start and stop time specifications are returned.

Input: Valid combinations of data set start/stop times for the search request.

Output: Returned search results package.

Expected Results: The returned search results package is verified for content.

Test Case 2 Invalid Data Set Start/Stop Time Format

This test verifies that a search request cannot be constructed using invalid formats for the data set start/stop time search criteria. The tester attempts to enter values outside the defined ranges for start and stop times. The system responds with an invalid format alarm.

Input: Invalid combinations of data set start/stop time format for the search request.

Output: Invalid format alarms.

Expected Results: Alarm messages for invalid search criteria format are received for each attempt to enter an invalid format.

Test Case 3 Invalid Data Set Start/Stop Time Period

This test verifies that invalid time periods are detected. The tester attempts to enter a negative time period (stop time less than start time). Also, the tester enters a period for which their is no data (future time and time before data was collected). The system is able to respond to the invalid time periods prior to initiating a search request.

Input: Negative time periods, time periods outside of valid collection period.

Output: Alarms indicating invalid periods detected.

Expected Results: Alarm messages are generated for each attempt to input an invalid time period as well as an alarm stating the requested time period is outside of the data collection period.

Test Case 4 Valid Start/Stop Revolution Time Combinations

This test verifies data search request results from using the start/stop revolution time format. The tester inputs various combinations of start and stop revolution times and inspects the returned results package to ensure that only data within the start and stop revolution time specifications are returned.

Input: Valid combinations of start/stop revolution times for the search request.

Output: Returned search results package.

Expected Results: The returned search results package is verified for content.

Test Case 5 Invalid Start/Stop Revolution Time Format

This test verifies that a search request cannot be constructed using invalid formats for the start/stop revolution time search criteria. The tester attempts to enter values outside the defined ranges for start and stop times. The system responds with an invalid format alarm.

Input: Invalid combinations of start/stop revolution time format for the search request.

Output: Invalid format alarms.

Expected Results: Alarm messages for invalid revolution time format is received for each attempt to enter an invalid format.

Test Case 6 Invalid Start/Stop Revolution Time Period

This test verifies that invalid time periods are detected. The tester attempts to enter a negative time period (stop time less than start time). Also, the tester enters a period for which their is no data (future time and time before data was collected). The system is able to respond to the invalid time periods prior to initiating a search request.

Input: Negative time periods, time periods outside of valid collection period.

Output: Alarms indicating invalid periods detected.

Expected Results: Alarm messages are generated for each attempt to input an invalid time period as well as an alarm stating the requested time period is outside of the data collection period.

5.1.8.2.10 Sequence 10 - Building a Complex Query Test

Another option available to users is the use of the logical operands "and", "not", and "or" in building complex queries to perform data searches. The following tests exercise several options of complex queries.

Test Case 1 Complex query using "AND"

The tester inputs a complex query composed of multiple "and's". After the query is built, it is examined and submitted. The search results package is verified. The result package only contains character data in response to the search request, no graphical data is available.

Input: Complex query composed with multiple "and" operators, expected results, search request.

Output: Search result package.

Expected Results: The search result package matches the expected results.

Test Case 2 Complex query using "OR"

The tester inputs a complex query composed of multiple "or's". After the query is built, it is examined and submitted. The search results package is verified. The result package only contains character data in response to the search request, no graphical data is available.

Input: Complex query composed with multiple "or" operators, expected results, search request.

Output: Search result package.

Expected Results: The search result package matches the expected results.

Test Case 3 Complex query using "NOT"

The tester inputs a complex query composed of multiple "not's". After the query is built, it is examined and submitted. The search results package is verified. The result package only contains character data in response to the search request, no graphical data is available.

Input: Complex query composed with multiple "not" operators, expected results, search request.

Output: Search result package.

Expected Results: The search result package matches the expected results.

Test Case 4 Multiple combination test

The tester inputs a complex query composed of multiple "and's", "or's", and "not's". After the query is built, it is examined and submitted. The search results package is verified. The result package only contains character data in response to the search request, no graphical data is available.

Input: Complex query composed with multiple operators, expected results, search request.

Output: Search result package.

Expected Results: The search result package matches the expected results.

Test Case 5 Negative Complex Query

The tester inputs a complex query that excludes every data set in the system. The query is composed of multiple "not's" and "and's" in order to exclude everything. The search result package contain the appropriate messages to indicate that no data matched the search criteria.

Input: Complex query composed of multiple negative operators, search request.

Output: Search result package.

Expected Results: The search result package contain a message indicating no data was found.

Test Case 6 All Encompassing Complex Query

The tester inputs a complex query that include every data set registered at a LIM. The query is composed of multiple operators in order to include everything. The search result package contain all data sets registered at the LIM. The result package only contains character data in response to the search request, no graphical data is available.

Input: Complex query composed of multiple operators, search request.

Output: Search result package.

Expected Results: The search result package contain all data sets.

5.1.8.2.11 Sequence 11 - Dependent Valids Test

One option in database query construction is choosing items from lists of options. Dependent valids refers to the process of narrowing the available options. For example, if a user selects a certain platform while constructing a query and then wishes to select sensors on that platform, the list to chose from only contains those sensors which reside on the selected platform. The following tests verify that this process of dependent valids is functional.

Test Case 1 Source/Platform and Sensor/Instrument Test

The tester selects a platform(s) from a list in order to build a query. After a platform is selected, the tester then moves to the sensor/instrument list. The available

sensors/instruments are available on the selected platform. The tester needs to prepare a list of dependent valids in order to verify the test results. After verifying the list of sensor/platforms, the tester makes a selection and submits a request. The returned result package is verified to insure the data matches the query specifications. The result package is represented graphically.

Input:	List of valid platform/instrument combinations, search request.
Output:	Search result package.
Expected Results:	The search result package contain only data available on the selected platforms and instruments.

Test Case 2 Campaign/Project and Geophysical Test

The tester selects a campaign/project(s) from a list in order to build a query. After a project is selected, the tester then moves to the geophysical list. The available geophysical's relate to the selected project. The tester needs to prepare a list of dependent valids in order to verify the test results. After verifying the list of geophysical's, the tester makes a selection and submits a request. The returned result package is verified to insure the data matches the query specifications. The result package is represented graphically.

Input:	List of valid project/geophysical combinations, search request.
Output:	Search result package.
Expected Results:	The search result package contain only data available on the selected projects and geophysical's.

Test Case 3 Entire List

For this test the tester constructs a query using every available list. The tester needs to prepare a list of dependent valids containing information from every available list. After the query is created and the dependent valids have been verified, the search request is submitted. The returned result package is verified to insure the resulting data fits the search criteria. The result package is represented graphically.

Input:	List of dependent valids containing selections from every available list of options, search request.
Output:	Search result package.
Expected Results:	The search result package matches the final search criteria.

Test Case 4 Graphical Dependent Valid Search

The tester constructs a search request utilizing the various graphical tools available. First, the tester selects a global area of map using the mapping tool. The tester then refines the selected region using the bounding box tool with the draw option. This box is drawn smaller than the global region. The tester then uses the point/radius tool to draw a circle within the box. This circle further refines the selected region. The tester then goes to the time-line tool. At this point, a graphical representation of the time periods for the selected regions are displayed. The test stops at this time and the graph is verified. The

tester needs to know at this time what data is available for the region that was selected. Now, using the time line graphical selection tool, the tester further refines the search to select only a portion of the total data. The search request is then submitted and the returned search result package is verified.

Input: Search request, Valid list of region/time oriented data.

Output: Search result package.

Expected Results: The search result package contain all data for the region/time specified in the search request.

Test Case 5 Character Dependent Valid Search

The tester constructs a search request utilizing the various search selections available. The tester first selects a source/platform. After selecting the source/platform, the tester next moves to the sensor/instrument box. Only those sensor/instruments available on the source/platform selected are available for selection. The tester chooses the desired sensor/instruments. The tester next goes to the campaign/project box and again only those campaign/projects with data on the selected sensor/instruments on the source/platforms selected are available. The tester then goes to the geophysical box to verify the selectable items are being refined to reflect previous search criteria entered. After verifying the remaining lists of options, the tester submits the request and validates the returned search result package.

Input: Search request, Valid list of dependent valid options.

Output: Search result package.

Expected Results: The search result package contain all data which matches the search criteria selected.

Test Case 6 Graphical/Character Dependent Valid Search

The tester constructs a search request utilizing both the various search selections available and the graphical tools. The tester first uses the mapping tool to select a region to begin the search criteria. The tester next selects a source/platform. After selecting the source/platform, the tester next moves to the sensor/instrument box. Only those sensor/instruments available on the source/platform selected are available for selection. The tester chooses the desired sensor/instruments. The tester next goes to the campaign/project box and again only those campaign/projects with data on the selected sensor/instruments on the source/platforms selected are available. The tester then goes to the geophysical box to verify the selectable items are being refined to reflect previous search criteria entered. After verifying the remaining list of options, the tester selects the time line tool. A graphical representation of the available data on the time line is displayed. The tester selects the desired time period from the graph. The tester then returns to the mapping tool, where a graphical representation of the available data matching the search criteria is displayed. The tester then submits the request and validate the returned search result package.

Input: Search request, Valid list of dependent valid options.

Output: Search result package.

Expected Results: The search result package contain all data which matches the search criteria selected.

5.1.8.2.12 Sequence 12 - CERES Data Search by Keyword

This sequence verifies that CERES data can be searched on by the following fields: product ID, sensor name, and data center. The tester first generates a CERES data search list. The returned result package contains all the CERES data available. Then, each test case narrows the list by performing a search on the keyword. The returned result package is then verified for completeness of data by keyword. The test case also ensures no extraneous data was provided.

Test Case 1 CERES Data List Search by Product ID

This test case verifies that CERES data can be searched on by product ID. The tester generates a data search on CERES data specifying a product ID keyword. After the search result package has been received, the tester prints the data list and verify that the CERES data contains only data related to the product ID provided and that no extraneous data was included.

Input: CERES data search request by product ID.

Output: CERES data search result package.

Expected Results: The data list contains all the CERES data information related to the product ID keyword.

Test Case 2 CERES Data List Search By Sensor Name

This test case verifies that CERES data can be searched on by sensor name. The tester generates a data search on CERES data specifying a sensor name keyword. After the search result package has been received, the tester prints the data list and verifies that the CERES data contains only data related to the sensor name provided and that no extraneous data was included.

Input: CERES data search request by sensor name.

Output: CERES data search result package.

Expected Results: The data list contains all the CERES data information related to the sensor name keyword.

Test Case 3 CERES Data List Search By Data Center

This test case verifies that CERES data can be searched on by data center. The tester generates a data search on CERES data specifying a data center keyword. After the search result package has been received, the tester prints the data list and verifies that the CERES data contains only data related to the data center provided and that no extraneous data was included.

Input: CERES data search request by data center.

Output: CERES data search result package.

Expected Results: The data list contains all the CERES data information related to the data center keyword.

Test Case 4 Invalid CERES Data List Search

This test case verifies that if an invalid search criteria is entered, that the proper error messages are displayed. Not all search criteria are available for all data. The tester attempts to perform a search on a search criteria not applicable to CERES data. This action results in an error message indicated the search field is not applicable to CERES data.

Input: Invalid CERES data keyword search request.

Output: CERES data lists, alarm message indicating invalid search criteria.

Expected Results: The system generates an alarm message indicating an invalid keyword search criteria was selected.

5.1.8.2.13 Sequence 13 - TRMM Data Search by Keyword

This sequence verifies that TRMM data can be searched on by the following fields: granule ID, start date/time, stop date/time, sensor ID, and latitudinal/longitudinal resolution. The tester first generates a TRMM data search list. The returned result package contain all the TRMM data available. Then, each test case narrows the list by performing a search on the keyword. The returned result package is then verified for completeness of data by keyword. The test case also ensures no extraneous data was provide.

Test Case 1 TRMM Data Keyword Search by Granule ID Keyword

This test case verifies that TRMM data can be searched on by granule ID. The tester generates a data search on TRMM data specifying a granule ID keyword. After the search result package has been received, the tester prints the data list and verifies that the TRMM data contains only data related to the granule ID provided and that no extraneous data was included.

Input: TRMM data search request by granule ID.

Output: TRMM data search result package.

Expected Results: The data list contains all the TRMM data information related to the granule ID keyword.

Test Case 2 TRMM Data Search By Start Date/Time Keyword

This test case verifies that TRMM data can be searched on by start date/time. The tester generates a data search on TRMM data specifying a start date/time keyword. After the search result package has been received, the tester prints the data list and verifies that the TRMM data contains only data related to the start date/time provided and that no extraneous data was included.

Input: TRMM data search request by start date/time.

Output: TRMM data search result package.

Expected Results: The data list contains all the TRMM data information related to the start date/time keyword.

Test Case 3 TRMM Data Search By Stop Date/Time Keyword

This test case verifies that TRMM data can be searched on by stop date/time. The tester generates a data search on TRMM data specifying a stop date/time keyword. After the search result package has been received, the tester prints the data list and verifies that the TRMM data contains only data related to the stop date/time provided and that no extraneous data was included.

Input: TRMM data search request by stop date/time.

Output: TRMM data search result package.

Expected Results: The data list contains all the TRMM data information related to the stop date/time keyword.

Test Case 4 TRMM Data Search By Sensor ID Keyword

This test case verifies that TRMM data can be searched on by sensor ID. The tester generates a data search on TRMM data specifying a sensor ID keyword. After the search result package has been received, the tester prints the data list and verifies that the TRMM data contains only data related to the sensor ID provided and that no extraneous data was included.

Input: TRMM data search request by sensor ID.

Output: TRMM data search result package.

Expected Results: The data list contains all the TRMM data information related to the sensor ID keyword.

Test Case 5 TRMM Data Search By Latitudinal/Longitudinal Resolution Keyword

This test case verifies that TRMM data can be searched on by latitudinal/longitudinal resolution. The tester generates a data search on TRMM data specifying a latitudinal/longitudinal resolution keyword. After the search result package has been received, the tester prints the data list and verifies that the TRMM data contains only data related to the latitudinal/longitudinal resolution provided and that no extraneous data was included.

Input: TRMM data search request by latitudinal/longitudinal resolution.

Output: TRMM data search result package.

Expected Results: The data list contains all the TRMM data information related to the latitudinal/longitudinal resolution keyword.

Test Case 6 Invalid TRMM Data Search

This test case verifies that if an invalid keyword search criteria is entered, that the proper error messages are displayed. Not all keyword search criteria are available for all data.

The tester attempts to perform a keyword search by TBD keyword. This action results in an error message indicated the search field is not applicable.

Input: Invalid TRMM data search request.

Output: Alarm message indicating invalid search criteria.

Expected Results: The system generates an alarm message indicating an invalid keyword search criteria was selected.

5.1.8.2.14 Sequence 14 - CERES Data Sort by Keyword

This sequence verifies that when CERES data is received after a search request that it can be sorted on the various fields of product ID, sensor name, and data center.

Test Case 1 CERES Data List Sort by Product ID

This test case verifies that CERES data can be sorted by product ID. The tester generates a data search on CERES data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by product ID. The data list is then sorted. The tester verifies that the sorted list includes all data from the original list, sorted by product ID.

Input: CERES data search request, sort request.

Output: CERES data lists, one sorted by data type, one sorted by product ID.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 2 CERES Data List Sort By Sensor Name

This test case verifies that CERES data can be sorted by sensor name. The tester generates a data search on CERES data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by sensor name. The data list is then sorted. The tester verifies that the sorted list includes all data from the original listed, sorted by sensor name.

Input: CERES data search request, sort request.

Output: CERES data lists, one sorted by data type, one sorted by sensor name.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 3 CERES Data List Sort By Data Center

This test case verifies that CERES data can be sorted by data center. The tester generates a data search on CERES data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests

the data to be sorted by data center. The data list is then sorted. The tester verifies that the sorted list includes all data from the original listed, sorted by data center.

Input: CERES data search request, sort request.

Output: CERES data lists, one sorted by data type, one sorted by data center.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 4 Invalid CERES Data List Sort

This test case verifies that if an invalid sort criteria is entered, that the proper error messages are displayed. Not all sort criteria are available for all data. The tester requests and receives a CERES data list. The tester then attempts to perform a sort by TBD. This action results in an error message indicating the sort field is not applicable.

Input: CERES data search request, invalid sort request.

Output: CERES data lists, alarm message indicating invalid sort criteria.

Expected Results: The system generates an alarm message indicating an invalid sort criteria was selected.

5.1.8.2.15 Sequence 15 - TRMM Data Sort by Keyword

This sequence verifies that when TRMM data is received after a search request that it can be sorted on the various fields of granule ID, start date/time, stop date/time, sensor ID, latitudinal/longitudinal resolution.

Test Case 1 TRMM Data List Sorted by Granule ID

This test case verifies that TRMM data can be sorted by granule ID. The tester generates a data search on TRMM data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by granule ID. The data list is then sorted. The tester verifies that the sorted list includes all data from the original list, sorted by granule ID.

Input: TRMM data search request, sort request.

Output: TRMM data lists, one sorted by data type, one sorted by granule ID.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 2 TRMM Data List Sorted by Start Date/Time

This test case verifies that TRMM data can be sorted by start date/time. The tester generates a data search on TRMM data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by start date/time. The data list is then sorted. The tester

verifies that the sorted list includes all data from the original list, sorted by start date/time.

Input: TRMM data search request, sort request.

Output: TRMM data lists, one sorted by data type, one sorted by start date/time.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 3 TRMM Data List Sorted By Stop Date/Time

This test case verifies that TRMM data can be sorted by stop date/time. The tester generates a data search on TRMM data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by stop date/time. The data list is then sorted. The tester verifies that the sorted list includes all data from the original list, sorted by stop date/time.

Input: TRMM data search request, sort request.

Output: TRMM data lists, one sorted by data type, one sorted by stop date/time.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 4 TRMM Data List Sort By Sensor ID

This test case verifies that TRMM data can be sorted by sensor ID. The tester generates a data search on TRMM data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by sensor ID. The data list is then sorted. The tester verifies that the sorted list includes all data from the original list, sorted by sensor ID.

Input: TRMM data search request, sort request.

Output: TRMM data lists, one sorted by data type, one sorted by sensor ID.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 5 TRMM Data List Sorted By Latitudinal/Longitudinal Resolution

This test case verifies that TRMM data can be sorted by latitudinal/longitudinal resolution. The tester generates a data search on TRMM data, requesting it to be sorted by data type. After the search result package has been received, the tester prints the data list. The tester then requests the data to be sorted by latitudinal/longitudinal resolution. The data list is then sorted. The tester verifies that the sorted list includes all data from the original list, sorted by latitudinal/longitudinal resolution.

Input: TRMM data search request, sort request.

Output: TRMM data lists, one sorted by data type, one sorted by latitudinal/longitudinal resolution.

Expected Results: Both data lists contain the same information regardless of the order sorted.

Test Case 6 Invalid TRMM Data List Sort

This test case verifies that if an invalid sort criteria is entered, that the proper error messages are displayed. Not all sort criteria are available for all data. The tester requests and receives a TRMM data list. The tester then attempts to perform a sort by TBD. This action results in an error message indicated the sort field is not applicable.

Input: TRMM data search request, invalid sort request.

Output: TRMM data lists, alarm message indicating invalid sort criteria.

Expected Results: The system generates an alarm message indicating an invalid sort criteria was selected.

5.1.8.2.16 Sequence 16 - Coincidence Data Viewing

This sequence verifies that a returned search result package can be displayed in various formats at the same time. Each test case views the same search result package using a different COTS/public domain user interactive tool. The tester originates a search request and submit it. Once the search result package is returned, the tool is used to represent the data in various ways. Each method utilized is assessed to ensure that all data is represented.

Test Case 1 Display Footprints

This test case views the returned search result package as footprints of the images superimposed upon a regional map. The map representation is verified to ensure all data in the returned search result package is accounted for.

Input: Search request, returned search result package.

Output: Search result package data images superimposed upon a regional map.

Expected Results: The map representation contains all the data that was in the original search result package.

Test Case 2 View Details

This test case views details, at all available levels, for the returned search result package. Data is verified at each level of detail.

Input: Search request, returned search result package.

Output: Search result package reviewed by detail.

Expected Results: Each level of detail contains all the data that was in the original search result package.

Test Case 3 Alpha-numeric Table Representation

This test case converts the returned result package into an alpha-numeric table. The table is verified to ensure all data is accounted for.

Input: Search request, returned search result package.

Output: Search result package arranged in an alpha-numeric table.

Expected Results: All data in the original search result package is accounted for and represented in alpha-numeric order.

5.1.9 Data Access Services Thread (T11-03)

5.1.9.1 Thread Objectives

The objectives of the data access services thread are as follows:

- Provide the capability to manipulate data residing in the data dictionary (i.e., define new ECS definition) and to manipulate the schema (i.e., add new field to existing data dictionary data base) of the data dictionary. Capability to save data dictionary information for backup purposes is provided and recovery procedures verified.
- Provide the capability to access the data dictionary to obtain information about data objects, data elements, terms, etc.
- Provide the capability to manipulate the data defined in the LIM schema.
- Provide the capability to access data and services on heterogeneous data servers at a single ECS site (LIM).
- Provide the capability to access the advertising service to perform searches to locate a service and install the service on the users scientist workbench.
- Provide the capability to directly access information contained in the guide, directory, and inventory level of the data server.

5.1.9.2 Thread Test Description

Testing is performed to demonstrate the ability to manipulate (add, modify, and delete) data defined in the data dictionary. Testing will demonstrate the system does not accept and attempt to add an invalid to the data dictionary. Verification that only a administrator of the data dictionary can make manipulations is performed.

Testing is performed to verify the ability to change the schema of the data dictionary. Adding and deleting of fields defined in the data dictionary is performed. Testing will verify that existing fields within the data dictionary can not be modified.

Testing will demonstrate the ability to access information in the data dictionary. Attempts to access data that is not contained in the data dictionary is performed to verify the system provides a response to the requester in this situation.

Testing will verify the ability to maintain the data defined in the LIM schema. Specific search criteria's are performed to demonstrate the ability to access the data contained in the LIM to determine the location of a particular set of data.

Testing to verify the accessibility of the advertising service is performed. Testing will verify the capability to install a particular service found in the advertiser onto the local scientist workbench.

Dependencies: (If Applicable)

In order to verify the objectives of the data access services thread, the following interface and functional capabilities are required:

- Write permissions to the defined data dictionary.
- Ability to change the schema of the data dictionary.
- Ability to write to external media to perform backup saves of the data dictionary.
- Ability to access the data dictionary using views and contexts. Direct access to the top levels of the data server.
- Heterogeneous data server environment within a local site.
- LIM schema defined for heterogeneous data server environment. Write permissions to allow changes to the LIM schema.
- Ability to access the LIM schema to determine data locations.
- Existence of an advertising service that contains a list of ECS services supported in Rel A. Ability to access the advertising service from a local workstation.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  -
- Data:

5.1.9.2.1 Sequence 1 - Re-Configure the Data Dictionary

The following tests verify the administrator's capabilities to manipulate the data in the data dictionary (add, delete, modify entries). Testing is performed to verify that invalid entries for the data dictionary are rejected.

<u>Test Case 1</u> <u>Add Context Information to DD</u>

This test verifies the capability of entering context based summary data for a new data type/product in the DD. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. Then, the tester enters summary information for a new data type in the DD.

Input: Context based summary information, administration account and DDS administration tool.

Output: The data dictionary summary information is updated. Administration log contains record of new summary item added to DD.

Expected Results: The new summary information is properly entered and is accessible by the user community.

Test Case 2 Add an Entry in DD

This test verifies the capability of adding an individual entry to the data dictionary. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. The tester then enters information for a new dictionary entry in the data dictionary.

Input: Data dictionary information, administration account and DDS administration tool.

Output: The data dictionary contains a new entry. Administration log contains record of new entry being added to the DD.

Expected Results: The new item is properly entered into the DD and is accessible by all users.

Test Case 3 Attempt to add an Invalid Summary

This test verifies that invalid summary information will not be entered in the data dictionary. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. The tester then attempts to enter a summary for a new data type in a incorrect format in the data dictionary. The tester then closes the administrative application.

Input: Invalid summary information, administration account.

Output: Error message concerning attempt to add data using an invalid format. Administration log contains record of attempt to add and invalid entry to the DD.

Expected Results: The system does not allow the information to be added to the data dictionary when its in an invalid format.

Test Case 4 Attempt to add an Invalid Entry in DD

This test verifies the capability of rejecting the addition of an individual entry with invalid format in the data dictionary. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. Next, the tester attempts to enter information, with an invalid format, for a data dictionary entry.

Input: Invalid data dictionary entry information, administration account.

Output: Error message concerning attempt to add data using an invalid format. Administration log contains record of attempt to add and invalid entry to the DD.

Expected Results: The system does not allow the information to be added to the data dictionary when its in an invalid format.

Test Case 5 Modify an Entry in DD

This test verifies the capability to modify an individual entry in the data dictionary. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. Modifies an existing entry in the data dictionary.

Input: Data dictionary information to be modified, administration account, and administration tool.

Output: The data dictionary item is modified. Administration log contains record of the item that was modified in the DD.

Expected Results: The item was modified in the DD and is accessible by all users.

Test Case 6 Modify Multiple Entries in DD

This test verifies the capability to modify multiple entries in the data dictionary. The tester (acting as an administrator of the DD) invokes an application for administration of the DDS. The tester then modifies several entries in the DD.

Input: Data dictionary information to be modified, administration account, and administration tool.

Output: The data dictionary items are modified. Administration log contains record of the item that was modified in the DD.

Expected Results: The items are modified in the DD and is accessible by all users.

Test Case 7 Delete an Entry in DD

This test verifies the capability to delete an individual entry in the data dictionary. The tester (acting as the administrator of the DD) invokes an application for administration of the DDS. The tester then deletes an entry in the data dictionary.

Input: Data dictionary information, administration account and DDS administration tool.

Output: The data dictionary entry is deleted. Administration log contains record of the deleted entry.

Expected Results: The entry is deleted in the DD and no longer accessible by the users.

Test Case 8 Media Backup of Data Dictionary

This test verifies the capability to save the data dictionary to external media. The tester (acting as the administrator of the DD) invokes an application for administration of the

DDS. The tester selects an option to perform external backup save of the data dictionary. Each external media backup option is tested (i.e., 4mm, 8mm, CD-ROM)

Input: Data dictionary information, administration account and DDS administration tool. Option to select 'external backup media save' of the data dictionary.

Output: The data dictionary is saved to the selected external media. Administration log contains an entry for the execution of the external save.

Expected Results: Data saved to external media is the same as the data on-line. Backup of data dictionary information can be performed.

Test Case 9 Recovery of Data Dictionary

This test verifies the capability to recover the data dictionary upon disk failure. While accessing the data dictionary, the tester induces a disk failure. Recovery procedures are initiated to restore the data dictionary from the backup created in test case 8. The user is able to complete the interrupted operation after the recovery is complete.

Input: Data dictionary transaction, tester induced disk failure, recovery procedures.

Output: The disk failure is discovered, alarms generated, event logs updated.

Expected Results: The data dictionary recovery procedures are executed and the initiated transaction is completed after recovery procedures are executed successfully.

5.1.9.2.2 Sequence 2 - Monitor Data Dictionary Services

The following test verifies that a non-administrator can not perform updates to the data dictionary.

Test Case 1 DD Privileges

This test verifies that anyone attempting to make updates (add, modify, and delete) to the DD accept the DD administrator is denied access. Log on as a non-data dictionary administrator and attempt to bring up the DD tool.

Input: Non-data dictionary administrator logon id.

Output: User interface provided does not contain the data dictionary administrator tool. Non-administrator can not manipulate the data dictionary. Event log contains an entry for the user logon.

Expected Results: Data dictionary tool is not presented on the desktop for the user to attempt to access.

5.1.9.2.3 Sequence 3 - Change Data Dictionary Schema

The following tests verify the administrator's capabilities to change the schema of the data dictionary.

Test Case 1 Add a new Field

This test verifies the capability of adding a new field to the existing schema of the DD.

Input: Log onto the ECS system as an administrator. Invoke an application for administration of the DDS. Add a new field (i.e., employee phone numbers) to the existing schema of the DD.

Output: The new field is added to the existing schema of the DD. Field has been properly 'blank' filled for all existing entries in the data dictionary (i.e., will not contain invalid values). Administration log contains an entry for the change to the data dictionary schema.

Expected Results: The new field is accessible for data entry. No data exists in the new field for any entries already in the data dictionary.

Test Case 2 Delete an existing Field

This test verifies the capability of deleting an existing field from the schema of the DD.

Input: Log onto the ECS system as an administrator. Invoke an application for administration of the DDS. Delete an existing field (i.e., employee phone numbers) from the existing schema of the DD.

Output: The deleted field is removed from the existing schema of the DD. Attempts to access entries in the removed field result in an error condition. Administration log contains an entry for the change to the data dictionary schema.

Expected Results: The deleted field is no longer accessible for data entry.

Test Case 3 Modify an existing Field

This test verifies that an existing field in the data dictionary can not be modified.

Input: Log onto the ECS system as an administrator. Invoke an application for administration of the DDS. Modify an existing field from one title to another and change the field size.

Output: Administrator provided with an error message indicating that field names and sizes can not be changed. Attempt to change field fails. No changes are made to the data dictionary schema. Administration log contains an entry for the attempt to change the data dictionary schema.

Expected Results: Data dictionary schema unchanged.

5.1.9.2.4 Sequence 4 - Accessing Information that exists in the Data Dictionary

The following tests verify the capability to access information in the data dictionary from the scientist workbench. Testing is performed to verify that the system provides the user with a response to a query for non-existent data.

Test Case 1 Access DD using Views

This test verifies the capability of accessing the DD using data dictionary views. Views restrict the access to a specific subset of the data dictionary information. This test case is executed for various subset categories available in the data dictionary (i.e., product names, product id, station id, etc).

Input: Bring up the scientist workbench, activate the tool to access the data dictionary, and query the data dictionary for a list of product names.

Output: User requested search was restricted and only data that fit into the category specified in the search (i.e., product names) is displayed. Data access log is updated.

Expected Results: List of product names provided. Event log contains entries for query initiated and results provided. Data access log is updated.

Test Case 2 Access DD using Contexts

This test verifies the capability of accessing the DD using data dictionary contexts. Access requests to the DD can reference a context and thus restrict the visibility of data dictionary information to that context. This test case is executed for various context categories available in the data dictionary (i.e., atmospheric dynamics, L0, L1, etc).

Input: Bring up the scientist workbench, activate the tool to access the data dictionary, and query the data dictionary for a list of items/data that pertain to atmospheric dynamics.

Output: User requested search was restricted and only data that fit into the category specified in the search (i.e., atmospheric dynamics) is displayed. Data access log is updated.

Expected Results: List of information that relates to atmospheric dynamics is provided. Event log contains entries for query initiated and results provided. Data access log is updated.

Test Case 3 Access DD for Non-Existent Data

This test verifies that the system provides the user with a message indicating that the data requested was not found when a query for non-existent data is received.

Input: Query for a known non-existent data product.

Output: Status message provided to the user indicating the data requested does not exist on the system. Event log contains entries for query initiation and result.

Expected Results: System generates a response to the data query for non-existent data.

5.1.9.2.5 Sequence 5 - Maintaining the LIM Schema

The following series of tests verify the administrator's ability to maintain the LIM schema. The system's file directories are updated for the transactions in this sequence.

Test Case 1 Add New Data to LIM Schema

This test verifies the administrator's capability to add new data to the LIM schema. After the data is added to the LIM, an attempt is made to access the data.

Input: Administrator tool, add of new data type, new product, etc.

Output: New data type, etc has been added to the LIM schema and is accessible to the user community. Administration log contains an entry for the change to the LIM schema. File directory is updated.

Expected Results: The LIM database contains the updated information.

Test Case 2 Delete Data from LIM Schema

This test verifies the administrator's capability to delete existing data from the LIM schema. After the data is deleted from the LIM, an attempt is made to access the data.

Input: Administrator tool, deletion of an existing data set and its associated information.

Output: The data set is deleted, the LIM schema is updated and the deleted data set is no longer accessible by the user community. Administration log contains an entry for the change to the LIM schema. File directory is updated.

Expected Results: The LIM database no longer has the data set information.

Test Case 3 Update Existing Data in LIM schema

This test verifies the administrator's capability to update existing data in LIM schema. After the data is updated in the LIM, an attempt is made to access the data.

Input: Administrator tool, modification to an existing data set, etc.

Output: The modification is made, the LIM schema is updated and the updated data is accessible to the user community. Administration log contains an entry for the change to the LIM schema. File directory is updated.

Expected Results: The LIM database contains the updated information.

5.1.9.2.6 Sequence 6 - LIM Data Searches

A LIM is an analogous hierarchy of data dictionary information which allows access operations to a heterogeneous data server environment at a single site. The following series of tests verify

the ability to access the LIM data dictionary information. Testing is performed to verify that the system provides the user with a response to a query for non-existent data.

Test Case 1 Accessing Data via a LIM

This test verifies the ability to access the information in the data dictionary for a LIM environment. Access to a particular sites data should be available through an interface on the scientist workbench. Testing is performed to demonstrate the ability to directly access each DAAC supported in Rel A.

Input: Scientist workbench. Select option to search the data dictionary for the DAAC under test. Generate a query for a known data product for the DAAC under test.

Output: Data products for the DAAC under test are provided to the tester. Event log contains entries of logon to the DAAC under test and the query results. Data access log is updated.

Expected Results: Only data available for the DAAC under test is provided. Data access log is updated.

Test Case 2 Access DD for Non-Existent Data

This test verifies that the system provides the user with a message indicating that the data requested was not found when a query for non-existent data is received.

Input: Scientist workbench. Select option to search the data dictionary for the DAAC under test. Generate a query for a non-existent data product for DAAC under test.

Output: Status message provided to the user indicating the data requested does not exist on the system. Event log contains entries for query initiation and result.

Expected Results: System generates a response to the data query for non-existent data.

5.1.9.2.7 Sequence 7 - Locating/Installing Advertising Services to the Scientist Workbench

The following series of tests demonstrate the ability to access the advertising services via the scientist workbench and install new services to the local workstation. Testing is performed to produce a failure in the attempt to install a service to verify the system provides the user with status messages for the failure.

Test Case 1 Determining Available Advertising Services

This test verifies that the user is able to access the advertising services from the scientist workbench and 'browse' a list of services available.

Input: Scientist workbench. Advertising services. Tester selects option to 'browse' the list of services that are available to the ECS community.

Output: Tester is automatically connected to the advertising service and is provided with a list of services that are available on the ECS system. Event log contains entry for activation of advertising services connection.

Expected Results: Tester is able to connect to advertising services from the scientist workbench and obtain a list of the services that are available.

Test Case 2 Installing A New Service

This test verifies that the user is able to access the advertising services from the scientist workbench and install a new service to the local workbench.

Input: Scientist workbench. Advertising services. Tester browses the list of services that are available to the ECS community and selects a service to install on the local workbench.

Output: Service selected is installed and displayed as an icon on the local workbench. Selecting the icon results in a direct connection to the service. Event log contains entries for the connection to the advertising service, the selection of the service to be installed, the successful installation of the service, and the activation of the service after it was installed on the local workbench.

Expected Results: Tester is able to connect to the advertising service and install a new service to the local workbench from the scientist workbench.

Test Case 3 Failure Installing A New Service

This test verifies that the system provides the user with status messages and error conditions when problems occur attempting to install a new service to the local workbench.

Input: Modified scientist workbench to 'write protect' the entire workbench (workbench set to 'read' only at the highest level to prohibit writing of data to the workbench). Tester browses the list of services that are available to the ECS community and selects a service to install on the local workbench.

Output: Service selected is not installed to the local workbench. Status message indicating an error occurred during the attempt to install the service to the local workbench. Event log contains entries for the activation of the advertising service and the error occurred during the attempt to install the service to the local workbench.

Expected Results: Tester is able to connect to the advertising service and receives an error message when an error occurs during an attempt to install a new service to the local workbench..

5.1.9.2.8 Sequence 8 - Accessing Top Levels of the Data Server

The following tests verify the ability to directly access the top levels of the data server (directory, guide, and inventory) from the scientist workbench. The system's file directory is accessed during the transactions in this sequence.

Test Case 1 Accessing Directory Level Information

This test verifies the capability to access the directory information. The directory is a collection of uniform descriptions that summarize the contents of a large number of data sets. The scope of information at this layer is very broad.

Input: Log onto the ECS system. Using the scientist workbench interface, select option to view the directory level of information on the local data server.

Output: The directory level information for the local data server is presented to the user. Data access log is updated.

Expected Results: The user is presented with a broad list of information contained in the directory of the local data server. Data access log is updated.

Test Case 2 Guide Level Information

This test verifies the capability to access the guide information. The guide is a detailed description of a number of data sets and related entities containing information suitable for making a determination of the nature of each data set and its potential usefulness for a specific application. The scope is of medium breadth, covering datasets held within EOSDIS.

Input: Log onto the ECS system. Using the scientist workbench interface, select option to view the guide level of information on the local data server.

Output: The guide level information for the local data server is presented to the user. Data access log is updated.

Expected Results: The user is presented with a detailed description of data sets contained in the guide of the local data server. Data access log is updated.

Test Case 3 Inventory Level Information

This test verifies the capability to access the inventory information. The inventory is a uniform set of descriptions of granules from one or more data sets with information required to select and obtain a subset of those granules. The primary use of this layer view is for local, regional, or global product searches returning enough information in the results to identify specific granules and allow further investigation, or to determine that it is suitable to access.

Input:	Log onto the ECS system. Using the scientist workbench interface, select option to view the inventory level of information on the local data server.
Output:	The inventory level of information for the local data server is presented to the user. Data access log is updated.
Expected Results:	The user is presented with a uniform set of descriptions of granules from one or more data sets contained in the inventory of the local data server. Data access log is updated.

5.1.10 System Integration Build 2 (B11)

The System Integration Build 2 is an integrated build of the following system threads:

- Enhanced System Access Thread
- CSMS Security Thread
- Client Services Thread
- Search Services Thread
- Data Access Services Thread

5.1.10.1 Build Objectives

The objectives of the system integration build 2 are:

- Provide secure access to ECS desktop through Bulletin Boards and the internet.
- Provide the means for first time users to register within the ECS system.
- Provide ECS desktop and workbench services.
- Provide access to the EOSView tool for data visualization.
- Provide a method to search the data server guide, directory, and inventory table.
- Provide advertising service to perform searches to locate and install services on the scientist workbench.
- Provide data access methods to include data dictionary, LIM schema, and data servers.
- Provide the ability for authorized users to modify access control lists, and add/modify user accounts.

5.1.10.2 Build Test Description

Testing is performed to verify the user environment supports the EOSDIS data services. The ECS desktop and workbench services are verified. Using these services, the tester generates data search scenarios. The data visualization tool, EOSView, is used to visualize returned search result packages. Testing is also performed, to ensure the advertising service is operational. Access to the data dictionary, LIM schema, data servers, and security management by authorized personnel is also verified.

Dependencies

In order to verify the objectives of the system integration build 2, the following interface and functional capabilities are required:

- Network environment (LAN and WAN).
- User registration, scientist workbench, and EOSView.
- Data server containing CERES & TRMM simulated data.
- Functional data dictionary.
- Functional LIM.
- Access to methods used to build search criteria.
- Interface to data server, data dictionary, and LIM for data access.
- Interface to EOSView for data visualization.
- Interface to Advertising Service.
- Access to data dictionary summary administration tool for modifying data dictionary.
- Access to administration tool for modifying LIM schema.
- Access to security administration tool for editing authorization privileges.

Test Support Requirements

- Hardware:
  –
- Software:
  –
- Data:
  –

5.1.10.2.1 Sequence 1 - Accessibility and Registration

The following series of tests demonstrates the ability for first time users or guest users to access the ECS system and register for system privileges, through the use of bulletin boards or internet services. These tests will verify the functionality of communication services, system security, and management services.

<u>Test Case 1</u> <u>Bulletin Board Access</u>

This test demonstrates the ability of a first time user or guest user to access the ECS system through bulletin board services. This test will also verify the ECS desktop will be displayed to the tester's display and inform them to enter a username and password. The tester will use the guest account to access the ECS desktop and proceed to complete the registration process. Tester access and registration is recorded and verified.

Input: Logon to ECS system as a guest user via bulletin board services. Complete registration "form".

Output: Guest username authenticated for access onto ECS system from bulletin board services.

Expected Results: Guest username is authenticated and is allowed access to scientist workbench. System services are limited to guest authorization privileges. Security log is printed to verify completeness of security services.

Test Case 2 Direct Internet Access

This test demonstrates the ability of a first time user or guest user to access the ECS system through direct internet services. This test will also verify the ECS desktop will be displayed to the users display and inform the user to enter a username and password. The tester will use the guest account to access the ECS desktop and proceed to complete the registration process. Tester access and registration is recorded and verified.

Input: Internet address for ECS system. Logon to ECS system as a guest user via direct internet access. Complete registration "form".

Output: ECS informational desktop is displayed to tester's machine. Tester allowed access through guest username.

Expected Results: Guest username is authenticated and tester is allowed access to scientist workbench. System services are limited to guest privileges. Security log is printed to verify completeness of security services.

Test Case 3 TCP/IP Access

This test demonstrates the ability of a first time user or guest user to access the ECS system through TCP/IP services. This test will also verify the ECS desktop will be displayed to the users display and inform the user to enter a username and password. The tester will use the guest account to access the ECS desktop and proceed to complete the registration process. Tester access and registration is recorded and verified.

Input: Internet address for ECS system. Logon to ECS system as a guest user via TCP/IP access. Complete registration "form".

Output: ECS informational desktop is displayed to tester's machine. Tester allowed access through guest username.

Expected Results: Guest username is authenticated and tester is allowed access to scientist workbench. System services are limited to guest privileges. Security log is printed to verify completeness of security services.

Test Case 4 GOSIP Access

This test demonstrates the ability of a first time user or guest user to access the ECS system through GOSIP services. This test will also verify the ECS desktop will be displayed to the users display and inform the user to enter a username and password. The tester will use the guest account to access the ECS desktop and proceed to complete the registration process. Tester access and registration is recorded and verified.

Input: Internet address for ECS system. Logon to ECS system as a guest user via GOSIP access. Complete registration "form".

Output: ECS informational desktop is displayed to tester's machine. Tester allowed access through guest username.

Expected Results: Guest username is authenticated and tester is allowed access to scientist workbench. System services are limited to guest privileges. Security log is printed to verify completeness of security services.

Test Case 5 Registration through e-mail

This test verifies the tester is able to register for ECS system services through e-mail services.

Input: E-mail message requesting registration application for ECS services. Completed application.

Output: ECS service application sent to requesting tester, tester returns completed application back to ECS.

Expected Results: Depending on user location, user class, or data preferences, an application from a corresponding DAAC will be sent to the user. Upon ECS receipt of the completed application, registration will be processed and response sent to requesting user.

Test Case 6 Registration through BBs services

This test verifies that a tester is able to register through BBs services for ECS access and services.

Inputs: Through BBs link, ECS registration application is completed and transmitted to ECS system.

Outputs: ECS service application received and processed.

Expected Outputs: Application received at appropriate DAAC (dependent on tester location, data preferences, or user class), processed and response sent to requesting tester.

5.1.10.2.2 Sequence 2 - Accountability

The following series of tests verify the ability for a registered and guest user to logon to the system and perform a data request. The tester will attempt to access both private and public data types within the guest account, to verify system authorization privileges. The functionality of communication services, system security, and management services will also be verified.

Test Case 1 Registered User Logon - Accountability

This test verifies that a registered user is able to logon to the system and obtain a particular data set or product. Testing will verify user authorization to obtain the data set or product. The data set or product requested will be transferred to the user via ftp to

verify communication services. All transactions will be logged in the event log and reviewed/printed to verify the functionality of the management services.

Input: Logon to ECS system as a known registered user via the scientist workbench. Locate a data set or product that is accessible to the login id used and transfer the data via ftp to the local workbench.

Output: Data set or product is transferred to the local scientist workbench and can be displayed by the user. Login and transfer of data to the local workbench is recorded in the event log.

Expected Results: Registered user able to log directly onto the ECS system without being prompted to complete a registration form. User access to the data set or product is verified prior to the data being transferred to the local workstation. Data is transferred via ftp to verify communication services. Event log is printed to verify completeness of management services (system accountability) of transactions being performed on the system.

Test Case 2 Guest User Logon - Public Data Access

This test verifies that a guest user is able to logon to the system and obtain a particular data set or product within the public domain. Testing will verify user authorization to obtain the data set or product. The data set or product requested will be transferred to the user via RPC to verify communication services. All transactions will be logged in the event log and reviewed/printed to verify the functionality of the management services.

Input: Logon to the ECS system as a guest user and download the scientist workbench to the local workstation. Bring up the scientist workbench and logon to the ECS system. Locate a data set or product, within the public inventory, that is accessible to the login id used and transfer the data via RPC to the local workbench.

Output: Data set or product is transferred to the local scientist workbench and can be displayed by the user. Login and transfer of data to the local workbench is recorded in the event log.

Expected Results: Guest user able to log onto the ECS system and receive requested data set or product on local workstation.

Test Case 3 Guest User Logon - Non-Public Data Access

This test verifies that a guest user is unable to access data that is not within the public domain, once logged into the ECS system. Testing will verify user authorization, and all transactions will be logged in the event log and reviewed/printed to verify security access.

Input: Logon to the ECS system as a guest user and bring up the scientist workbench. Locate a data set or product within the non-public domain and attempt to transfer the data to the local workstation.

Output: Tester receives error message stating the guest account does not have access to requested data. Login and data request is recorded in the event log.

Expected Results: Guest user able to log onto the ECS system. Account is not authorized to access data within non-public volumes, error message is displayed to tester.

Test Case 4 Multiple User - Accountability

This test verifies that correct logistical data is collected when multiple users are accessing the system simultaneously. Variations of accessibility to the system are performed by each user (i.e., data search, data request, data transfer). Verification of each users accessibility to data (if the user requested data or is attempting to transfer data) is performed. All transactions performed by each user is logged in the event log. The event log is printed to verify that management services is logging the correct transactions for each user.

Input: Multiple registered and guest users logon to the ECS system. All users perform a variety of tasks (i.e., data search, data request, data transfer, registration, browsing of data).

Output: All users obtain search results, data products, etc. based on function performed. Login and activities performed by each user are recorded in the event log. Any data files transferred are logged in the transaction log.

Expected Results: User(s) access the data set or product is verified prior to the data being transferred to the local workstation. Data is transferred via ftp, RPC, or other electronic means to verify communication services. Event log is printed to verify completeness of management services (system accountability) of multiple transactions being performed.

5.1.10.2.3 Sequence 3 - Client Interaction

The following series of tests verify the ability to use the ECS desktop and workbench services to locate, install, and execute services, from SCFs, DAACs, and ECS, to/from the workbench. The tester, using the workbench, locates a service and loads it on the workbench. From the workbench, the tester launches the application and performs a search and/or browse and then views the returned data using EOSView on the local workstation. This test case is executed to test the various options of services available from the advertising service (i.e., browse services, search services, etc.).

Test Case 1 Obtaining a Service

This test verifies that a client is able to access the advertising service, locate a particular service, install it to the local workbench, and then execute the application from the workbench.

Input: ECS workbench, advertising service, and a service in the advertiser (i.e., browse service, search service).

Output: Service selected is installed on the local workbench. Client is able to activate the service installed on the workbench and perform functions available from the application. Event log contains an entry for the activation of the workbench, access to the advertising service, installation of the service to the local workbench, and activation of the application from the workbench.

Expected Results: Drag and drop of the service from the advertiser to the workbench is successful. User is able to activate application after installation and perform desired operations.

Test Case 2 Accessing the Data Dictionary

This test verifies that a client is able to access the data dictionary to aid the user in generating a data search request. Utilizing this information, user generates a search request for a particular data set and obtains a results package. User then views the result package using EOSView.

Input: ECS desktop, access to the data dictionary, search request. EOSView.

Output: Client accesses the data dictionary to gain knowledge to support the generation of a search request. Search request generated and submitted. Search results package received by the client. Client reviews search result package utilizing EOSView. Event log contains an entry user access to the data dictionary, generation of the search request, receipt of search results package, and activation of EOSView to view the result package.

Expected Results: Client is able to access the data dictionary to aid in formulating the search request, generates a search request and receives a search result package for the request. User is able to view the results using EOSView.

Test Case 3 Accessing the Data Server

This test verifies that a client is able to directly query the data server for a data set or product.

Input: ECS desktop, access to the data server, search request. EOSView.

Output: Client submits a request for a data product directly to the data server to obtain a known data product. Search package results received by the client. Client reviews search package result utilizing EOSView. Event log contains an entry user access to the data server, generation of the search request, receipt of search results package, and activation of EOSView to view the result package.

Expected Results: Client is able to submit a search request directly to the data server to obtain a known data product. Client receives a search package

result for the request. User is able to view the results using EOSView.

Test Case 4 Accessing the LIM

This test verifies that a client is able to generate a search request to the LIM to locate a data set in a heterogeneous data server environment.

Input: ECS desktop, search request to be processed by the LIM, EOSView.

Output: Client submits a request for a data product to the LIM. Search package results received by the client. Client reviews search package result utilizing EOSView. Event log contains an entry user access to the data server, generation of the search request, receipt of search results package, and activation of EOSView to view the result package.

Expected Results: Client is able to submit a search request to the LIM. Client receives a search result package for the request. User is able to view the results using EOSView.

Test Case 5 Accessing DAAC-Unique Searching Services

This test verifies that a client is able to access local DAAC user interfaces, through the ECS workbench, to perform local metadata searching services.

Input: ECS desktop, access to local DAAC user interface. EOSView.

Output: Through use of ECS desktop functions, tester access DAAC user interface and submits a request for a data product local to the DAAC. Client reviews search package result utilizing EOSView. Event log contains an entry for tester access to local DAAC IMS, generation of the search request, receipt of search results package and activation of EOSView to view the result package.

Expected Results: Tester is able to access local DAAC IMS through ECS desktop. Tester authorized to perform metadata search through DAAC IMS, and receives a search result package. Tester is able to view the results using EOSView.

5.1.10.2.4 Sequence 4 - Data Search Services

The following series of tests verify the ability to use the ECS desktop and workbench services to perform a search for data products within the data server guide, directory, and inventory tables using different methods of search techniques. From the workbench the tester launches the search service and performs a search and/or browse and then views the returned data using EOSView on the local workstation. This test sequence is executed to test the various methods of search techniques available within the search service of the ECS desktop.

Test Case 1 Registered User - Graphical Directory Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and using the graphical directory search, construct and submit a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package is viewed using EOSView.

Input: Registered username and password, ECS workbench, data sets controlled by a LIM or data server, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 2 Registered User - Guide Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a graphical or textual guide search. The tester will utilize the on-line help and data dictionary to construct and submit a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, graphical or textual guide search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 3 Registered User - Graphical Inventory Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and using the graphical inventory search tools (mapping, time-line, lat/lon line, etc.), constructs and submits a land or ocean data request for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid land/ocean data region, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 4 Registered User - Character Inventory Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and using the character inventory search, constructs and submits a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid character fields, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 5 Registered User - Spatial Metadata Search - Global Granule

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a graphical inventory search using global land/ocean attributes. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, mapping tool, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 6 Registered User - Spatial Metadata Search - Point/Radius

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a graphical inventory search using a point/radius search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, circular area of interest, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 7 Registered User - Spatial Metadata Search - Bounding Box

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a graphical inventory search using a bounding box search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, rectangular area of interest, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 8 Registered User - Graphical Core Metadata Search (Time)

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a core metadata search using a time index search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, data period input for time line tool, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 9 Registered User - Graphical Core Metadata Search (Lat/Lon)

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a core metadata search using a mapping tool. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, mapping tool, search request.

Output: Successful login to ECS system. Search result package. Image/animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 10 Registered User - Character Core Metadata Search (Algo)

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on algorithm information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid algorithm information including: algorithm package acceptance date, algorithm package description, algorithm intended operating system, algorithm package maturity code, algorithm package name, algorithm package version, and algorithm test data description, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 11 Registered User - Character Core Metadata Search (Author/Date/Instrument)

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on documentation information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid search information including: author information, date of reference, instrument short name, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 12 Registered User - Character Core Metadata Search (Vertical/Horizontal)

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on vertical/horizontal information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid search information including: region description, region name, spatial format, spatial coverage, and spatial resolution, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 13 Registered User - Time Coincident Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a data search using a time coincident search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, valid time represented in year/month/day, hour/minute/second, or Julian format, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 14 Registered User - Start/Stop Time Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a data search using data set start and stop time and start/stop revolution time searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Registered username and password, ECS workbench, data set start/stop times or revolution start/stop times, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 15 Registered User - Complex Query Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a data search by building complex queries using logical operands. The search request is submitted for data accessible to the login id. The search

result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Registered username and password, ECS workbench, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 16 Registered User - Dependent Valids Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a data search using dependent valid searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Registered username and password, ECS workbench, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 17 Registered User - CERES Keyword Data Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a CERES data search using associated keyword searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Registered username and password, ECS workbench, valid product id, sensor name, or data center, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 18 Registered User - TRMM Keyword Data Search

This test verifies that a registered user is able to access the search service from the ECS workbench, and perform a TRMM data search using associated keyword searches. The search request is submitted for data accessible to the login id. The search result package

is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Registered username and password, ECS workbench, valid granule id, start/stop date/time, sensor id, or lat/lon resolution search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 19 Guest User - Graphical Directory Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and using the graphical directory search, construct and submit a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package is viewed using EOSView.

Input:	Guest account, ECS workbench, data sets controlled by a LIM or data server, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 20 Guest User - Guide Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a graphical or textual guide search. The tester will utilize the on-line help and data dictionary to construct and submit a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, graphical or textual guide search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 21 Guest User - Graphical Inventory Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and using the graphical inventory search tools (mapping, time-line, lat/lon line, etc.), constructs and submits a land or ocean data request for data accessible to the

login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, valid land/ocean data region, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 22 Guest User - Character Inventory Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and using the character inventory search, constructs and submits a data request for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, valid character fields, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 23 Guest User - Spatial Metadata Search - Global Granule

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a graphical inventory search using global land/ocean attributes. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, mapping tool, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 24 Guest User - Spatial Metadata Search - Point/Radius

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a graphical inventory search using a point/radius search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, circular area of interest, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 25 Guest User - Spatial Metadata Search - Bounding Box

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a graphical inventory search using a bounding box search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, rectangular area of interest, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 26 Guest User - Graphical Core Metadata Search (Time)

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a core metadata search using a time index search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, data period input for time line tool, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 27 Guest User - Graphical Core Metadata Search (Lat/Lon)

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a core metadata search using a mapping tool. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, mapping tool, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 28 Guest User - Character Core Metadata Search (Algo)

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on algorithm information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, valid algorithm information including: algorithm package acceptance date, algorithm package description, algorithm intended operating system, algorithm package maturity code, algorithm package name, algorithm package version, and algorithm test data description, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 29 Guest User - Character Core Metadata Search (Author/Date/ Instrument)

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on documentation information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, valid search information including: author information, date of reference, instrument short name, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 30 Guest User - Character Core Metadata Search (Vertical/Horizontal)

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a core metadata search using a textual search on

vertical/horizontal information. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, valid search information including: region description, region name, spatial format, spatial coverage, and spatial resolution, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 31 Guest User - Time Coincident Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a data search using a time coincident search. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, valid time represented in year/month/day, hour/minute/second, or Julian format, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 32 Guest User - Start/Stop Time Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a data search using data set start and stop time and start/stop revolution time searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input:	Guest account, ECS workbench, data set start/stop times or revolution start/stop times, search request.
Output:	Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.
Expected Results:	Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 33 Guest User - Complex Query Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a data search by building complex queries using logical operands. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 34 Guest User - Dependent Valids Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a data search using dependent valid searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 35 Guest User - CERES Keyword Data Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a CERES data search using associated keyword searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, valid product id, sensor name, or data center, search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

Test Case 36 Guest User - TRMM Keyword Data Search

This test verifies that a guest user is able to access the search service from the ECS workbench, and perform a TRMM data search using associated keyword searches. The search request is submitted for data accessible to the login id. The search result package is verified against the search criteria and the result package will be viewed using EOSView.

Input: Guest account, ECS workbench, valid granule id, start/stop date/time, sensor id, or lat/lon resolution search request.

Output: Successful login to ECS system. Search result package. Image/ animation is displayed in EOSView on the local workstation.

Expected Results: Tester is able to perform search request and search result package is transmitted to the local workstation. Data is displayed in EOSView on local workstation.

5.1.10.2.5 Sequence 5 - Data Dictionary Access

The following series of tests verify only authorized users are able to perform administration functions within the data dictionary. Administration functions include manipulating data residing in the data dictionary (adding, modifying, or deleting data). Testing includes authentication and authorization of administration account and access to data dictionary.

Test Case 1 Add Data in Data Dictionary - Administration

This test verifies the administration account has authorization to access data within the data dictionary and add new data.

Input: Administration account username and password. Data dictionary summary administration tool, entry data.

Output: The data dictionary contains a new entry. Administration log contains record of new entry being added to the data dictionary.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization account allows tester to add data to the data dictionary. Administration log records transaction information. New data is properly entered into data dictionary and is accessible by authorized users.

Test Case 2 Modify Data in Data Dictionary - Administration

This test verifies the administration account has authorization to access data within the data dictionary and modify entries.

Input: Administration account username and password. Data dictionary summary administration tool, data dictionary information to be modified.

Output: The data dictionary contains a modified entry. Administration log contains record of modification within data dictionary.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to modify data within the data dictionary. Administration log records transaction information. Modified data is properly entered into data dictionary and is accessible by authorized users.

Test Case 3 Delete Data in Data Dictionary - Administration

This test verifies the administration account has authorization to access data within the data dictionary and delete entries.

Input: Administration account username and password. Data dictionary summary administration tool, data dictionary information to be deleted.

Output: The data dictionary entry is deleted. Administration log contains record of deletion within data dictionary.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to modify data within the data dictionary. Administration log records transaction information. The entry is deleted in the data dictionary and is no longer accessible.

Test Case 4 Add Data in Data Dictionary - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the data within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to add data to the data dictionary.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, data dictionary information to be added.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to add data within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

Test Case 5 Modify Data in Data Dictionary - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the data within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to modify data within the data dictionary schema.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, data dictionary information to be modified.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to modify data within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

Test Case 6 Delete Data in Data Dictionary - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to delete data within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to delete data within the data dictionary schema.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, data dictionary information to be deleted.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to delete data within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

5.1.10.2.6 Sequence 6 - Schema Access

The following series of tests verify only authorized users are able to perform administration functions within the data dictionary and LIM. Administration functions include adding, modifying, and deleting existing fields within the schema. Testing includes authentication and authorization of administration account and access to data dictionary and LIM.

Test Case 1 Add New Field - Administration

This test verifies the administration account has authorization to access the data schema within the data dictionary and add a new field.

Input: Administration account username and password. Data dictionary summary administration tool, new field to add to the existing data schema.

Output: The data dictionary contains a new field. Administration log contains record of new field being added to the data dictionary schema.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to add data field to the data dictionary schema. Administration log records transaction information. New data field is accessible for data entry and is accessible by authorized users.

Test Case 2 Modify an Existing Data Schema- Administration

This test verifies the administration account has authorization to access the data schema within the data dictionary but is not allowed to modify existing data fields.

Input: Administration account username and password. Data dictionary summary administration tool, existing data field to modify.

Output: System responds with an error message indicating that field names can not be changed. No changes are made to the data dictionary schema. Administration log contains record of attempt to modify data fields within data dictionary.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account is not allowed to modify any data fields within data dictionary. Administration log records transaction information. No modifications are made to data dictionary schema.

Test Case 3 Delete an Existing Data Field - Administration

This test verifies the administration account has authorization to access the data schema within the data dictionary and delete existing data fields.

Input: Administration account username and password. Data dictionary summary administration tool, existing field within data schema to delete.

Output: The data field is deleted. Administration log contains record of deletion within data dictionary.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to delete data field within the data dictionary. Administration log records transaction information. The field is deleted within the data dictionary and is no longer accessible.

Test Case 4 Add a New Field - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the data schema within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to add a new data field within the data dictionary schema.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, new field to add to the existing data schema.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to add new data field to the data schema within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

Test Case 5 Modify an Existing Field - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the data schema within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to modify an existing data field within the data dictionary schema.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, existing field to modify within data schema.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to modify the data field within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

Test Case 6 Delete an Existing Field - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the data schema within the data dictionary. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to delete an existing data field within the data dictionary schema.

Input: Guest/Registered user account username and password. Data dictionary summary administration tool, existing data field to delete in the data schema.

Output: Tester is denied access to data dictionary summary administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to

delete existing data field within the data dictionary. Administration log records data dictionary access attempt. Data dictionary information is unchanged.

Test Case 7 — Add New Data to LIM Schema - Administration

This test verifies the administration account has authorization to add new data to the LIM schema.

Input: Administration account username and password. Administration tool to edit/modify LIM schema, new data type.

Output: The LIM schema contains the new data. Administration log contains record of new data being added to the LIM schema.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to add new data to the LIM schema. Administration log records transaction information. New data is accessible to all authorized users.

Test Case 8 — Delete Data from LIM Schema - Administration

This test verifies the administration account has authorization to delete data from the LIM schema.

Input: Administration account username and password. Administration tool to edit/modify LIM schema, deletion of existing data type.

Output: The data type is deleted from the LIM schema. Administration log contains record of data being deleted from the LIM schema.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to delete data from the LIM schema. Administration log records transaction information. The data is deleted from within the LIM schema is no login accessible.

Test Case 9 — Modify Data in LIM Schema - Administration

This test verifies the administration account has authorization to modify data in the LIM schema.

Input: Administration account username and password. Administration tool to edit/modify LIM schema, modification to existing data type.

Output: The LIM schema updated to contain the modified data. Administration log contains record of modification within the LIM schema.

Expected Results: Administration account is authenticated through the CSMS security system. Administration account authorization allows tester to modify data in the LIM schema. Administration log

records transaction information. New data is accessible to all authorized users.

Test Case 10 Add a New Data in LIM Schema - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the LIM schema. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to add new data within the LIM schema.

Input: Guest/Registered user account username and password. Administration tool to edit/modify LIM schema, new data to add to the LIM schema.

Output: Tester is denied access to the administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to add new data to the LIM schema. Administration log records data dictionary access attempt. LIM schema information is unchanged.

Test Case 11 Delete Data from LIM Schema - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to delete date from within the LIM schema. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to delete data from the LIM schema.

Input: Guest/Registered user account username and password. Administration tool to edit/modify LIM schema, existing data to delete from the LIM schema.

Output: Tester is denied access to the administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to delete data from within the LIM schema. Administration log records data dictionary access attempt. LIM schema information is unchanged.

Test Case 12 Modify Data in LIM Schema - Guest/Registered User

This test verifies the guest and registered user accounts do not have authorization to modify the LIM schema. Within this test the tester will access the ECS desktop as a guest or registered user and attempt to modify data within the LIM schema.

Input: Guest/Registered user account username and password. Administration tool to edit/modify LIM schema, existing data to modify within the LIM schema.

Output: Tester is denied access to the administration tool. Administration log contains record of attempted access by non-authorized account.

Expected Results: Guest/Registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to modify data within the LIM schema. Administration log records data dictionary access attempt. LIM schema information is unchanged.

5.1.10.2.7 Sequence 7 - Security Access

The following series of tests verify only authorized users are able to perform security administration functions. Security administration functions include adding, modifying, and deleting user accounts and modifying existing authorization privileges. Testing includes authentication and authorization of administration account and access to access control lists and user information.

Test Case 1 Adding Registered User - Administration

This test verifies that authenticated users are authorized to add/modify user information. Within this test a tester will access the ECS system under the administration account and add/edit existing registered user information.

Input: Administration account username and password. Security administration tool. User information to edit/add.

Output: New/modified user information is captured within file. Security audit log records addition/modification of user information.

Expected Results: Administration account is authenticated through the CSMS security system. Account authorization allows tester to add/modify user information within ECS system. Security audit log reflects tester modifications. User information updated and stored. Authentication and authorization meets GNMP standards.

Test Case 2 Deleting Registered User - Administration

This test verifies that authenticated users are authorized to delete a registered user. Within this test a tester will access the ECS system under the administration account and delete an existing registered user.

Input: Administration account username and password. Security administration tool. Existing registered user to delete.

Output: Registered user information is deleted from within the file. Security audit log records deletion of registered user.

Expected Results: Administration account is authenticated through the CSMS security system. Account authorization allows tester to delete user from ECS system. Security audit log reflects tester activities. User account is no longer existent on ECS system. Authentication and authorization meets GNMP standards.

Test Case 3 Adding Registered User - Guest/Registered User

This test verifies that guest and registered users are not authorized to add/modify user information. Within this test a tester will access the ECS system through a guest or registered user account and attempt to add/edit existing registered user information.

Input: Guest/registered user account username and password. Security administration tool. User information to edit/add.

Output: Tester is denied access to security administration tool. Security audit log records non-authorized attempt to access security administration tool.

Expected Results: Guest/registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to add/modify user information within ECS system. Security audit log reflects tester attempts. User information is unchanged. Authentication and authorization meets GNMP standards.

Test Case 4 Deleting Registered User - Guest/Registered User

This test verifies that guest and registered users are not authorized to delete a registered user. Within this test a tester will access the ECS system under the administration account and attempt to delete an existing registered user.

Input: Guest/registered user account username and password. Security administration tool. Existing registered user to delete.

Output: Tester is denied access to security administration tool. Security audit log records non-authorized attempt to access security administration tool.

Expected Results: Guest/registered user account is authenticated through the CSMS security system. Account authorization does not allow tester to delete user from ECS system. Security audit log reflects tester activities. User account is unchanged. Authentication and authorization meets GNMP standards.

5.1.10.2.8 Sequence 8 - User Interface

The following series of tests verify the capabilities of the ECS workbench. Capabilities include toolkit accessibility, portability, communication services, and interface extensions.

Test Case 1 Verification of Toolkit Access

When first delivered, the ECS workbench will contain toolkit services. This test verifies the accessibility of the delivered toolkit services. This test case is executed for each toolkit provided as a part of the initial ECS workbench configuration.

Input: Icon's on the workbench that provide the interface between the workbench and the supported toolkit. Select icon.

Output: The toolkit selected is presented to the user. User is able to enter data into the menu driven toolkit interface and receive valid data feedback.

Expected Results: Toolkit is accessible to the user. Tester is presented with a menu driven interface to the toolkit under test.

Test Case 2 Workbench/Toolkit Interaction

The test verifies that toolkit error or interrupt messages are interactively displayed to the tester's workbench. For this test, the tester will execute a toolkit service with incorrect data to receive an error message.

Input: Toolkit service. Data sets.

Output: Toolkit error message indicating data set incompatible with service.

Expected Results: Interaction between ECS workbench and toolkit service allows error message to be displayed to tester.

Test Case 3 Visualization Toolkit Portability

This test verifies that the data visualization toolkit is portable and able to execute on all ECS supported platforms.

Input: Data visualization service. Raster image file. Access to ECS supported platforms.

Output: Visualization service executes, and raster image is displayed on all ECS supported platforms

Expected Results: Data visualization service ported to all platforms.

Test Case 4 File Transfer

This test verifies that the ECS workbench has the capability to access the GSFC local IMS through remote log-on services, search for a data set and transfer the data set to the ECS system through a file transfer service.

Input: Guest/Registered user access to ECS workbench. Guest/Registered user access to GSFC local IMS. Search service from GSFC IMS on data set. File transfer, initiated from ECS workbench, to GSFC IMS on data set package.

Output: Tester successfully logs on to ECS and GSFC IMS. Transfer of data set from GSFC to ECS successful.

Expected Results: Tester authenticated for ECS workbench and GSFC local IMS. Account on GSFC IMS system authorized to access data sets searched. ECS workbench services allow tester to perform file transfer from GSFC local IMS. Security logs updated to reflect tester activities and file transfer.

Test Case 5 Mail Services

This test verifies that the ECS workbench contains mail service capabilities. To perform this test, the tester will access the ECS workbench and send mail to an account within the GSFC domain.

Input: Guest./registered user account access to ECS workbench. e-mail message sent to user within GSFC domain.

Output: Successful access to ECS workbench. e-mail service on ECS workbench. e-mail received at GSFC account.

Expected Results: Tester account authenticated for ECS workbench access. ECS workbench contains mail services, and tester account authorized for mail services. ECS system has access to internet, e-mail sent to account within GSFC domain. Security logs updated with tester access and activities.

5.1.10.2.9 Sequence 9 - System Performance

The following series of tests are for the purpose of analyzing the performance of the search services and user interfaces to determine if they meet desired performance.

Test Case 1 IMS Interface to DAAC Availability

Through modeling and analysis, this test will determine if the user interfaces to Information Management System (IMS) services at individual DAAC sites have an operational availability of .993 at a minimum and a mean down time of 2 hours.

Input: Model analysis reports.

Expected Results: Through modeling and analysis, IMS services at DAAC sites will have an operational availability of .9997 and a mean down time of 1.6 hours (design goals).

Test Case 2 Information Searches on ECS Directory

Through modeling and analysis, this test will determine if the architecture design for ECS will allow the function of information searches on the ECS directory to have an operational availability of .993 at a minimum and a mean down time of 2 hours or less.

Input: Model analysis reports.

Expected Results: Through modeling and analysis, information searches on the ECS directory will have an operational availability of .9997 and a mean down time of 1.4 hours. (Design goals).

Test Case 3 Information Searches on Local Data Holdings

Through modeling and analysis, this test will determine if the architecture design for ECS will allow the function of information searches on local data holdings to have an operational availability of 0.96 at a minimum and a mean down time of 4 hours or less.

Input: Model analysis reports.

Expected Results: Through modeling and analysis, information searches on local data holdings will have an operational availability of .999999 and a mean down time of 6 minutes. (Design goals).

5.1.11 Product Generation Thread (T12-01)

5.1.11.1 Thread Objectives

The objectives of the product generation thread are:

- To provide a method to read, create, and update the data availability schedule.
- To provide a method to read, create, delete, and modify process profile information.
- To provide a method to read, create, delete, and modify production strategies.
- To provide a method to read, create, delete, and modify data dependency rules.
- To provide a method to create, delete, modify, import and export a production plan.
- To provide a method to process auto initiate plans.
- To provide a method to perform on-demand processing requests for all defined data types for RMM (Rel A) (TRMM, CERES, and LIS).
- To provide a method to perform on-demand status requests for production plans in the queues for each defined data type process (TRMM, CERES, and LIS).
- To provide a method for reprocessing plans.
- To provide a method to order products.
- To provide a method to manipulate the processing plan queue.

5.1.11.2 Thread Test Description

This thread test verifies the ability to plan and manage the production planning environment. Testing is conducted to verify the capability to manipulate the data availability schedule, the process profile, and production strategies.

Testing is performed against the data dependency rules and standard processing rules. The tester will attempt to create, delete, and modify the standard processing rules. Rules are defined to control the production environment as it relates to data.

Testing is conducted to verify the ability to generate and manipulate production plans. This manipulation includes creating, deleting, modifying, importing and exporting plans. The production queues are loaded with plans to test the production queue management functions.

The tester will then test the systems timeline viewing tool used to manipulate the product planning queues graphically. This manipulation will include activating a plan, suspending a plan, resuming a plan, and canceling a plan.

The tester will verify the systems ability to process plan completion notices. The plan completion notice is sent to the plan originator upon plan completion and the product is available for archive.

Testing is performed to verify the systems ability to respond to stimuli in the production environment. The first test sequence demonstrates the auto initiate planning process of receiving and reacting to data availability notices. The test demonstrates the different environments a plan can be in when a data availability notice is received from the ingest service. Invalid notices are examined.

Testing is performed to demonstrate the capability to support on-demand processing requests. These requests are received outside of the normal processing environment to perform special processing. All variations of supported on-demand processing are tested.

Testing is performed to verify that on-demand status requests are processed. The requests are used to receive status updates of production plans in the system.

Testing is performed to verify the capability of performing reprocessing of previously processed data.

Dependencies: (If Applicable)

In order to verify the objectives of the production generation thread, the following interface and functional capabilities are required:

- Access to the functionality to manipulate the data availability schedule, the process profile information, the production strategies, and data dependency rules.
- Interface to the production queue for plan generation and manipulation.
- Interface to resource management for scheduling and execution.
- Test driver to simulate Data Availability Notices.
- The data server must contain data which can be used in the production generation process.
- Test drivers to simulate remote site requests for plans and products.

Test Support Requirements

- Hardware:
  - HP735, SUN Sparc10, DEC 3000/300, IBM RS6000, SGI R3000, and Cray
- Software:
  -
- Data:
  - Processing control parameter/resource files

5.1.11.2.1 Sequence 1 - Data Availability Schedule

This series of tests verifies that an operator can read, create, and update a data availability schedule. The data availability schedule is used in the generation of production plans and contains information concerning projected data availability.

<u>Test Case 1</u> <u>Reading a Data Availability Schedule</u>

This test case verifies that a user can read a Data Availability Schedule. The tester utilizes the user interface to select and read the schedule. The schedule is displayed on the screen.

Input:	Display data availability schedule request, data availability schedule.
Output:	The schedule is displayed.
Expected Results:	The data availability schedule is displayed upon operator request.

Test Case 2 Creating Data Availability Schedule

This test case verifies that an operator can create a data availability schedule. The tester uses the user interface to create a schedule of when data will be available. Message is logged concerning the creation of the schedule.

Input:	Create a data availability schedule message.
Output:	Message indicating the data availability schedule has been created.
Expected Results:	The tester is able to create a data availability schedule. A message indicating that the schedule has been created is logged.

Test Case 3 Insert New Information into a Data Availability Schedule

This test case verifies that a data availability schedule can be updated. The tester displays the schedule and utilizes the user interface to update the schedule. The update will insert new information into the schedule. The transaction is logged.

Input:	Update data availability schedule request, data availability schedule.
Output:	Information is updated in the schedule, log of changed information.
Expected Results:	Every update the tester makes to the data availability schedule is logged. The modified information is displayed on the new data availability schedule.

Test Case 4 Modify Existing Information in a Data Availability Schedule

This test case verifies that a data availability schedule can be updated. The tester displays the schedule and utilizes the user interface to update the schedule. The update will include modifying existing information in the schedule. All transactions are logged.

Input:	Update data availability schedule request, data availability schedule.
Output:	Information is updated in the schedule, log of changed information.
Expected Results:	Every update the tester makes to the data availability schedule is logged. The modified information is displayed on the new data availability schedule.

Test Case 5 Deleting Information in a Data Availability Schedule

This test case verifies that a data availability schedule can be updated. The tester displays the schedule and utilizes the user interface to update the schedule. The update will include deleting information from the schedule. All transactions are logged.

Input: Update data availability schedule request, data availability schedule.

Output: Information is updated in the schedule, log of changed information.

Expected Results: Every update the tester makes to the data availability schedule is logged. The modified information is displayed on the new data availability schedule.

Test Case 6 Invalid Updates to a Data Availability Schedule

This test case verifies that invalid updates to a data availability schedule are prohibited. The tester attempts to input invalid information into the schedule. The attempts are logged.

Input: Invalid updates to a data availability schedule.

Output: Information is not updated in the schedule, erroneous entry attempts are written to a log file.

Expected Results: As the tester attempts to input invalid information, the interface flags the inputs and does not update the schedule.

5.1.11.2.2 Sequence 2 - Process Profile Manipulation

This series of tests verifies that the process profile can be created, read, modified, and deleted. The information in the process profile is verified. The message logs are verified for success/failure messages.

Test Case 1 Process Profile Creation

This test case verifies that a valid request for creating a process profile is acted upon by the system. The operator enters a create a process profile message with the name of the process to be profiled. The system responds with a message stating that the profile has been created. The tester then displays the created profile. The tester then verifies the information in the profile. A "process profile created" message is logged.

Input: Create a process profile message, process name to create the profile for.

Output: The generated process profile for the requested process, message stating the profile is created, a display of the profile.

Expected Results: The system generates a process profile for the requested process. The information is displayed and validated by the tester. A message stating that the profile was generated is logged.

Test Case 2 Invalid Process Name for Process Profile Creation

This test case verifies that if an invalid process name is entered, that no process profile is generated.

Input: Create a process profile for a non-existing process.

Output: Message stating that an invalid process name was entered.

Expected Results: No profile is generated and a message is logged concerning the invalid request.

Test Case 3 Read a Process Profile

This test case verifies that the process profile can be read. The tester uses the user interface to read the process profile generated in test case 1. The profile is displayed and the information verified.

Input: Read a process profile message, process profile from test case 1.

Output: Process profile displayed.

Expected Results: The read a process profile request is accepted, validated, and the profile is displayed.

Test Case 4 Process Profile Modification

This test case verifies that the process profile can be modified. The tester uses the user interface to modify the parameters in the process profile. Parameters in the profile which can be modified include user requirements, costs, etc. The profile is displayed again and the changes are verified.

Input: Modify a process profile message, process profile from test case 1.

Output: Profile modified message.

Expected Results: The modify a process profile request is accepted, validated, and the changes are incorporated into the profile. A message indicating the profile has been modified is logged.

Test Case 5 Invalid Process Profile Modification

This test case verifies a process profile modification request is validated. The tester attempts to input invalid information into the profile (such as zero cost, latest completion time possible in the past, etc.) The invalid information is detected and error messages are generated. The profile is displayed again and the information is unchanged.

Input: Modify a process profile message with invalid information, process profile from test case 1.

Output: Messages indicating the update to process profile contains invalid information.

Expected Results: The modify a profile request is verified and errors encountered are reported in the message log. The profile is not altered.

Test Case 6 Process Profile Deletion

This test case verifies that the process profile created in test case 1 can be deleted. The tester uses the user interface to select and delete the process profile.

Input: Delete a process profile message, process profile from test case 1.

Output: Profile deleted message.

Expected Results: The delete a process profile request is accepted, validated, and the profile is deleted. A message indicating the profile has been deleted is logged.

Test Case 7 Invalid Process Profile Deletion

This test case verifies that the delete a process profile message is validated. The tester will attempt to delete a process profile which does not exist, or a profile for which the tester has no access to. The system responds with a message stating that the process profile cannot be deleted.

Input: Invalid process profile message.

Output: Profile not deleted message.

Expected Results: The delete a plan request is accepted, validated, and the erroneous information is discovered. A message indicating the profile has not been deleted is logged.

5.1.11.2.3 Sequence 3 - Production Strategies Manipulation

This series of tests verifies that the production strategies can be created, read, modified, and deleted. The information in the production strategy are verified. The message logs are verified for success/failure messages.

Test Case 1 Production Strategy Creation

This test case verifies that a valid request for creating a production strategy is acted upon by the system. The operator enters a create a production strategy message. The system responds with a message stating that the strategy has been created. The tester then displays the created production strategy. The tester then verifies the information in the strategy. A production strategy created message is logged.

Input: Create a production strategy message.

Output: The generated production strategy, message stating the strategy is created, a display of the strategy.

Expected Results: The system generates a production strategy. The information is displayed and validated by the tester. A message stating that the strategy was generated is logged.

Test Case 2 Read a Production Strategy Information

This test case verifies that a valid request for reading a production strategy is acted upon by the system. The operator utilizes the user interface to select and read production strategy information. The system responds with a display of the production strategy information.

Input:	Read production strategy information.
Output:	The selected production strategy information is displayed.
Expected Results:	The system displays the selected production strategy information.

Test Case 3 Production Strategy Modification

This test case verifies that the production strategy can be modified. The tester uses the user interface to modify the parameters in the production strategy. Parameters in the strategy which can be modified include processing priority, processing type, etc. The strategy is displayed again and the changes are verified.

Input:	Modify a production strategy message, production strategy from test case 1.
Output:	Strategy modified message.
Expected Results:	The modify a production strategy request is accepted, validated, and the changes are incorporated into the strategy. A message indicating the strategy has been modified is logged.

Test Case 4 Invalid Production Strategy Modification

This test case verifies a production strategy modification request is validated. The tester attempts to input invalid information into the strategy (such as unavailable resources, illegal priorities, etc.) The invalid information is detected and error messages are generated. The strategy is displayed again and the information is unchanged.

Input:	Modify a production strategy message with invalid information, production strategy from test case 1.
Output:	Messages indicating the update to production strategy contains invalid information.
Expected Results:	The modify a strategy request is verified and errors encountered are reported in the message log. The strategy is not altered.

Test Case 5 Production Strategy Deletion

This test case verifies that the production strategy created in test case 1 can be deleted. The tester uses the user interface to delete the production strategy.

Input:	Delete a production strategy message, production strategy from test case 1.
Output:	Strategy deleted message.

Expected Results: The delete a production strategy request is accepted, validated, and the strategy is deleted. A message indicating the strategy has been deleted is logged.

Test Case 6 Invalid Production Strategy Deletion

This test case verifies that the delete a production strategy message is validated. The tester will attempt to delete a production strategy which does not exist, or a strategy for which the tester has no access to. The system responds with a message stating that the production strategy cannot be deleted.

Input: Invalid delete a production strategy message.

Output: Strategy not deleted message.

Expected Results: The delete a plan request is accepted, validated, and the erroneous information is discovered. A message indicating the strategy has not been deleted is logged.

5.1.11.2.4 Sequence 4 - Data Dependency Rules

This series of tests verifies that data dependency rules can be entered, modified, and deleted from the product generation system. The tester attempts to add invalid rules. Data dependency rules are the guidelines the system uses in determining data availability for production planning.

Test Case 1 Entering Rules for Data Dependency

This test case verifies that operators can enter rules for data dependency. A report is generated prior to entering the rules. If the rules already exist, this test case deletes the rules, re-runs the report, and verifies that the rules were deleted. The tester then enters the rules for data dependency, re-runs the report and verify that the rules were entered into the system. A message is logged indicating the rules were entered.

Input: Report to verify if the rules exists, input of rules for data dependency.

Output: Report showing entered rules, message logged indicating rules entered.

Expected Results: The report generated reflects the entered rules.

Test Case 2 Modifying Rules for Data Dependency

This test case verifies that operators can modify the rules for data dependency. A report is generated prior to modifying the rules. The tester then modifies the rules for data dependency, re-runs the report and verifies that the rules were modified as indicated. A message is logged indicating the rules were modified.

Input: Report to verify the rules exists, input to modify the rules for data dependency, report to verify rules were modified.

Output: Report showing modified rules, message logged indicating rules were modified.

Expected Results: The report generated reflects the modified rules and a message is logged to indicate the rules were modified.

Test Case 3 Deleting Rules for Data Dependency

This test case verifies that operators can delete rules for data dependency. A report is generated to show the rules. This test case deletes the rules, re-runs the report, and verifies that the rules were deleted. The tester then generates the report again to verify the rules were deleted. A message is logged indicating the rules were deleted.

Input: Report to verify if the rules exist, input to delete the rules for data dependency.

Output: Report showing rules were deleted, message logged indicating rules deleted.

Expected Results: The report generated shows that the rules were deleted and a message is logged indicating this action.

Test Case 4 Entering Invalid Rules for Data Dependency

This test case verifies that operators cannot enter invalid rules for data dependency. A report is generated prior to entering the rules. The tester then attempts to enter invalid rules for data dependency, re-runs the report and verifies that the rules were not entered into the system. An example of invalid rules is attempting to declare that data is needed from a site where it does not exist, of that the L3 data is needed before L2 data is generated. Message is logged indicating the rules were invalid and not entered into the system.

Input: Report to verify existing rules, input of invalid rules for data dependency.

Output: Report showing rules were not entered, message logged indicating rules were invalid and not entered.

Expected Results: The system does not allow the invalid rules for data dependency to be entered.

5.1.11.2.5 Sequence 5 - Standard Processing Rules

This series of tests verifies that standard product rules can be entered, modified, and deleted from the product generation system. Attempts to add invalid rules are processed. Standard product rules are the guidelines the system uses in determining production planning.

Test Case 1 Entering Rules for Standard Processing

This test case verifies that operators can enter rules for standard processing. A report is generated prior to entering the rules. If the rules already exist, this test case deletes the rules, re-runs the report, and verifies that the rules were deleted. The tester then enters the rules for standard processing, re-runs the report and verifies that the rules were entered into the system. A message is logged indicating the rules were entered.

Input:	Report to verify if the rules exists, input of rules for standard processing.
Output:	Report showing entered rules, message logged indicating rules entered.
Expected Results:	The report generated reflects the entered rules.

Test Case 2 Modifying Rules for Standard Processing

This test case verifies that operators can modify the rules for standard processing. A report is generated prior to modifying the rules. The tester then modifies the rules for standard processing, re-runs the report and verifies that the rules were modified as indicated. A message is logged indicating the rules were modified.

Input:	Report to verify the rules exists, input to modify the rules for standard processing, report to verify rules were modified.
Output:	Report showing modified rules, message logged indicating rules were modified.
Expected Results:	The report generated reflects the modified rules and a message is logged to indicate the rules were modified.

Test Case 3 Deleting Rules for Standard Processing

This test case verifies that operators can delete rules for standard processing. A report is generated to show the rules. This test case deletes the rules, re-runs the report, and verifies that the rules were deleted. The tester then generates the report again to verify the rules were deleted. A message is logged indicating the rules were deleted.

Input:	Report to verify if the rules exist, input to delete the rules for standard processing.
Output:	Report showing rules were deleted, message logged indicating rules deleted.
Expected Results:	The report generated shows that the rules were deleted and a message is logged indicating this action.

Test Case 4 Entering Invalid Rules for Standard Processing

This test case verifies that operators cannot enter invalid rules for standard processing. A report is generated prior to entering the rules. The tester then attempts to enter invalid rules for standard processing, re-runs the report and verifies that the rules were not entered into the system. An example of invalid rules is attempting to generate L2 products in 30 seconds. Message is logged indicating the rules were invalid and not entered into the system.

Input:	Report to verify existing rules, input of invalid rules for standard processing.

Output: Report showing rules were not entered, message logged indicating rules were invalid and not entered.

Expected Results: The system does not allow the invalid rules for standard processing to be entered.

5.1.11.2.6 Sequence 6 - Routine Processing Plan Generation

This series of tests verifies that a processing plan can be created, modified, deleted, exported and imported by the ECS. The inputs to the request for plan manipulation are verified. The message logs are verified for success/failure messages as well as information passing between operational segments. A processing queue is generated by building additional plans.

Test Case 1 Process Plan Creation

This test case verifies that a valid request for creating a processing plan is acted upon by the system. The operator enters a create a plan message. The message must contain a start time, an end time, and planning criteria. Planning criteria must contain a processing algorithm. The system verifies the parameters and generates the plan.

Input: Create a plan message with valid start and finish times, and valid planning criterion.

Output: The system validates the plan and sends messages to operational segments.

Expected Results: Successful plan message is sent to the data server and the SMC, the plan and the associated metadata is sent to the data storage system.

Test Case 2 Invalid Start Time

This test case verifies that if a start time in the past is entered, that the system uses the current time for the start time of the plan.

Input: Create a plan message with a start time in the past, a valid finish time and valid planning criterion.

Output: Message adjusting start time, valid plan generated with appropriate messages sent to operational segments.

Expected Results: A successful plan message is sent to the data server and the SMC, the plan, with adjusted start time to current time, along with the associated metadata is sent to the data storage system.

Test Case 3 Invalid Finish Time

This test case verifies that when an invalid finish time is entered, that the system rejects the plan and the appropriate messages are sent.

Input: Create a plan message with a finish time in the past, a valid start time and valid planning criterion.

Output: Rejection message.

Expected Results: The create a plan request is rejected due to the invalid end time. The message indicating the error is logged.

Test Case 4 Time Period Validation - Too Short

This test case verifies that when a time period is too short to execute the plan, that the appropriate error messages will be displayed.

Input: Create a plan message with a time period which is too short for the amount of processing required, valid planning criterion.

Output: Rejection message or finish time adjustment.

Expected Results: The create a plan request is rejected due to the time period being too short to perform the processing or the finish time is adjusted. The message indicating the error is logged.

Test Case 5 Time Period Validation - Too Long

This test case verifies that when a time period is too long to execute the plan, that the plan is still generated.

Input: Create a plan message with a time period which is too long for the amount of processing required, a valid planning criterion.

Output: The plan is generated with a plan time within the start/finish window.

Expected Results: The create a plan request is processed within the time window requested.

Test Case 6 Invalid Time Period

This test case verifies that when an invalid time period is entered, that the system rejects the plan and the appropriate messages are sent.

Input: Create a plan message with a stop time less than the start time, and valid planning criterion.

Output: Rejection message.

Expected Results: The create a plan request is rejected due to the invalid time period. The message indicating the error is logged.

Test Case 7 Invalid Processing Algorithm

This test case verifies that when a create a plan request is received which contains a processing plan for an algorithm which does not exist, the plan is rejected.

Input: Create a plan message with an invalid processing algorithm and valid start and stop times.

Output: Rejection message.

Expected Results: The create a plan request is rejected due to an invalid algorithm criteria. The message indicating the error is logged.

Test Case 8 Data Type/Algorithm Mismatch

This test case verifies that a check is made between the data type requested for processing and the actual algorithm to be used. If the algorithm is not developed for use on the requested data type, an error message is generated and the plan is rejected.

Input: Create a plan message with an invalid data type/algorithm and valid start and stop times.

Output: Rejection message.

Expected Results: The create a plan request is rejected due to an invalid data type/algorithm combination. The message indicating the error is logged.

Test Case 9 Processing Plan Modification

This test case verifies that the plan created in test case 1 can be modified. The tester uses the user interface to modify the parameters in the processing plan. Parameters in the plan which can be modified include priority, time, or resource allocation.

Input: Modify a plan message, processing plan from test case 1.

Output: Plan modified message.

Expected Results: The modify a plan request is accepted, validated, and the changes are incorporated into the plan. A message indicating the plan has been modified is logged.

Test Case 10 Invalid Processing Plan Modification

This test case verifies a production plan modification request is validated. The tester attempts to input invalid information into the plan (such as erroneous data location information, invalid time parameters, etc.). The invalid information is detected and error messages are generated.

Input: Modify a plan message with invalid information, processing plan from test case 1.

Output: Plan contains invalid information message.

Expected Results: The modify a plan request is verified and errors encountered are reported in the message log. The plan is not altered.

Test Case 11 Processing Plan Deletion

This test case verifies that the plan created in test case 1 can be deleted. The tester uses the user interface to delete the processing plan.

Input: Delete a plan message, processing plan from test case 1.

Output: Plan deleted message.

Expected Results: The delete a plan request is accepted, validated, and the plan is deleted. A message indicating the plan has been deleted is logged.

Test Case 12 Invalid Processing Plan Deletion

This test case verifies that the delete a plan message is validated. The tester will attempt to delete a plan which does not exist, or a plan for which the tester has no access to. The system responds with a message stating that the plan cannot be deleted.

Input: Invalid delete a plan message.

Output: Plan not deleted message.

Expected Results: The delete a plan request is accepted, validated, and the erroneous information is discovered. A message indicating the plan has not been deleted is logged.

Test Case 13 Exporting a Processing Plan to the SMC

This test case verifies that the system can support the negotiation of production strategies between the local operations staff and the SMC staff. The tester will export a locally created plan (using test case 1) and export it to the SMC. The SMC verifies that the plan was received.

Input: Processing plan to export, export a plan to SMC message.

Output: Plan received by SMC message.

Expected Results: The export a plan request is accepted, validated, and the plan is exported to the SMC. The SMC acknowledges the plan was received.

Test Case 14 Importing a Processing Plan From the SMC

This test case verifies that once a plan was received by the SMC for conflict resolution, that the SMC can then modify the plan and send it back to the originator. The tester will await notification from the SMC concerning the plan sent to the SMC in test case 13. Once notification is received, the tester will import the plan from the SMC and incorporate the plan into the production queue.

Input: Notification of plan to import, processing plan to import from the SMC.

Output: Plan received from the SMC and placed into the production queue.

Expected Results: The plan previously exported in test case 13 is imported back to the originator. A message indicating the plan was received from the originator is logged.

Test Case 15 Production Queue Loading

This test case verifies that the production queue can be loaded with production plans. Test Case 1 of this sequence is repeated, with variant information, to load the production

queue with valid plans. This queue will then be tested in the next test sequence. The tester will verify that each plan is scheduled and placed in the queue correctly.

Input:	Create a plan messages with valid start and finish times, and valid planning criterion.
Output:	The system validates the plans and places the scheduled plans in the processing queues.
Expected Results:	Successful plan messages are sent to the data server and the SMC, the plans and the associated metadata are sent to the data storage system. The plans are placed in the production queue to await activation.

5.1.11.2.7 Sequence 7- Auto Initiate Processing

This series of tests verifies that when a data availability notice (DAN) is received from the ingest service that the notice is verified and that the proper actions occur upon receipt. Proper action is determined based on processing plan status. Invalid notices are examined. This sequence uses a test driver to simulate the DAN generation.

Test Case 1 Valid Data Availability Notice with Waiting Processing Plan

This test case verifies that when a valid DAN is received from the data server that the metadata is included. The tester first generates a processing plan for data that he has not ingested yet. The processing plan goes into a wait state for the data. The tester then ingests the data into the system. A message is then sent to the data processing system to indicate the data is available. The data availability notice is sent upon successful ingest of data into the ECS.

Input:	Processing plan which waits for the data, data to ingest, DAN with attached metadata.
Output:	Message logged to the data processing system.
Expected Results:	The DAN verifies the attached metadata and sends a message and the metadata to the data processing system for usage in the waiting processing plan.

Test Case 2 Valid Data Availability Notice with No Waiting Processing Plan

This test case verifies that when a valid DAN is received from the data server that the metadata is included. The system then checks to verify if a processing plan is waiting for this data. When no plan is found, a message is logged to indicate the data is available but has not been used. The data availability notice is sent upon successful ingest of data into the ECS.

Input:	Data availability notice with attached metadata.
Output:	Message logged indicating data ready but not used.
Expected Results:	The data availability notice verifies the attached metadata and log a message that the data is ready but has not been used.

Test Case 3 Invalid Data Availability Notice

This test case verifies that when a invalid DAN is received from the data server that the appropriate message is logged. An invalid DAN is one with no attached metadata. The data availability notice is sent upon successful ingest of data into the ECS.

Input: Data availability notice with no attached metadata.

Output: Invalid DAN message logged.

Expected Results: The data availability notice verifies the attached metadata and logs an error message when the metadata is not found.

5.1.11.2.8 Sequence 8 - On-Demand Processing Requests

This series of tests verifies that on-demand processing requests for all data types are processed as received. Three types of on-demand requests are verified; 1) process as soon as possible, 2) process within a specified time, or reject, and 3) process within a given time, or proceed anyway. Valid and invalid requests are examined.

Test Case 1 On-demand Processing - As Soon As Possible

This test case verifies that when an on-demand process request is received with a specification of "as soon as possible", that the system verifies the request and adds the plan to the current schedule in the first available window of opportunity. The plan is created for data which is known to be in the archives. A data query is sent to the data server to inquire about data availability. The resource schedule is verified for resource availability. The requested algorithm is verified as available for processing. The plan is then incorporated into the schedule based on resource and data availability. The tester verifies that the plan is inserted into the schedule at the first available opportunity. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with an as soon as possible parameter), local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled at first window of opportunity.

Expected Results: The plan is received and verified, the data query confirms that the data, resource, and algorithms are available, and the plan is scheduled for the first window of opportunity.

Test Case 2 On-demand Processing - As Soon As Possible - Data Unavailable

This test case verifies that when an on-demand process request is received, for data that is not in the system and with a specification of "as soon as possible", that the system rejects the request. The plan is created for data which is known not to be in the archives. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the data is

unavailable and the plan is rejected. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with an as soon as possible parameter) for non-existing data, local site schedule for data, resource and algorithm availability.
Output:	Data query to data server concerning data availability, plan rejection based on unavailable data.
Expected Results:	The plan is received and verified, the data query confirms that the data is unavailable, and the plan is rejected.
Test Case 3	On-demand Processing - As Soon As Possible - Resource Unavailable

This test case verifies that when an on-demand process request is received, for a resource which is unavailable and with a specification of "as soon as possible", that the system rejects the request. The plan is created to process on a resource which is unavailable. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the resource is unavailable and the plan is rejected. This test is repeated for resources defined to process data types for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with an as soon as possible parameter) for unavailable resource, local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data, resource, and algorithm availability, plan rejection based on unavailable resource requested.
Expected Results:	The plan is received and verified, the data query confirms that the resource is unavailable, and the plan is rejected.
Test Case 4	On-demand Processing - As Soon As Possible - Algorithm Unavailable

This test case verifies that when an on-demand process request is received, for an algorithm which is unavailable and with a specification of "as soon as possible", that the system rejects the request. The plan is created to process with an algorithm which is unavailable. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the algorithm is unavailable and the plan is rejected. This test is repeated for algorithms identified to process each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with an as soon as possible parameter) for unavailable algorithms, local site schedule for data, resource, and algorithm availability.

Output:	Data query to data server concerning data, resource, and algorithm availability, plan rejection based on unavailable algorithm requested.
Expected Results:	The plan is received and verified, the data query confirms that the algorithm is unavailable, and the plan is rejected.

Test Case 5 On-demand Processing - Within a Given Time or Reject

This test case verifies that when an on-demand process request is received with a specification of "within a given time or reject", that the system verifies the request and adds the plan to the current schedule within the specified time. The plan is created for data which is known to be in the archives. A data query is sent to the data server to inquire about data availability. The resource schedule is verified for resource availability during the specified time period. The requested algorithm is verified as available for processing during the requested time period. The plan is then incorporated into the schedule based on the requested time period. The tester verifies that the plan is inserted into the schedule as requested. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with a process within a given time or reject), local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled for requested time.
Expected Results:	The plan is received and verified, the data query confirms that the data, resource, and algorithms are available for the requested time period, and the plan is scheduled to execute during the requested time period.

Test Case 6 On-demand Processing - Within a Given Time or Reject - Data Unavailable

This test case verifies that when an on-demand process request is received, for data that is not in the system and with a specification of "process within a given time or reject", that the system rejects the request. The plan is created for data which is known not to be in the archives. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability during the available time. The data server responds that the data is unavailable now and is not scheduled to be in the system by the time period specified and the plan is rejected. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with a process within a given time or reject parameter) for non-existing data, local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data availability, plan rejection based on unavailable data.

Expected Results: The plan is received and verified, the data query confirms that the data is unavailable for the time period specified, and the plan is rejected.

Test Case 7 On-demand Processing - Within a Given Time or Reject - Resource Unavailable

This test case verifies that when an on-demand process request is received, for a resource which is unavailable and with a specification of "process within a given time or reject", that the system rejects the request. The plan is created to process on a resource which is unavailable during the specified time period. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the resource is unavailable during the specified time and the plan is rejected. This test is repeated for resources defined to process data types for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or reject parameter) for unavailable resource, local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data, resource, and algorithm availability, plan rejection based on unavailable resource requested.

Expected Results: The plan is received and verified, the data query confirms that the resource is unavailable, and the plan is rejected.

Test Case 8 On-demand Processing - Within a Given Time or Reject - Algorithm Unavailable

This test case verifies that when an on-demand process request is received, for an algorithm which is unavailable and with a specification of "process within a given time or reject", that the system rejects the request. The plan is created to process with an algorithm which is unavailable during the specified time period. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the algorithm is unavailable during the requested time period and the plan is rejected. This test is repeated for algorithms identified to process each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or reject parameter) for unavailable algorithms, local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data, resource, and algorithm availability, plan rejection based on unavailable algorithm during the requested time period.

Expected Results: The plan is received and verified, the data query confirms that the algorithm is unavailable, and the plan is rejected.

Test Case 9 On-demand Processing - Within a Given Time or Reject - Invalid Time Period

This test case verifies that when an on-demand process request is received, for an invalid time period and with a specification of "process within a given time or reject", that the system rejects the request. The plan is created with an invalid time period. The request parameters are verified and the invalid time period is detected. The plan is rejected based on the invalid time period. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or reject parameter) with an invalid time period.

Output: Invalid time period is detected, alarm message generated, plan rejected.

Expected Results: The plan is received and verified, the invalid time period is detected and the plan is rejected.

Test Case 10 On-demand Processing - Within a Given Time or Proceed Anyway

This test case verifies that when an on-demand process request is received with a specification of "within a given time or proceed anyway", that the system verifies the request and adds the plan to the current schedule within the specified time. The plan is created for data which is known to be in the archives. A data query is sent to the data server to inquire about data availability. The resource schedule is verified for resource availability during the specified time period. The requested algorithm is verified as available for processing during the requested time period. The plan is then incorporated into the schedule based on the requested time period. The tester verifies that the plan is inserted into the schedule as requested. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or proceed anyway), local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled for requested time.

Expected Results: The plan is received and verified, the data query confirms that the data, resource, and algorithms are available for the requested time period, and the plan is scheduled to execute during the requested time period.

Test Case 11 On-demand Processing - Within a Given Time or Proceed Anyway - Data Unavailable

This test case verifies that when an on-demand process request is received, for data that is not in the system and with a specification of "process within a given time or proceed

anyway", that the system schedule the processing when the data is scheduled to arrive. The plan is created for data which is known not to be in the archives but is scheduled to arrive at a time outside of the time specified in the request. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability during the available time. The data server responds that the data is unavailable now but returns the scheduled time of arrival. The plan is then inserted into the schedule based on the anticipated arrival time. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with a process within a given time or proceed anyway parameter) for non-existing data, local site schedule for data, resource and algorithm availability.
Output:	Data query to data server concerning data availability, plan scheduled based on scheduled data availability.
Expected Results:	The plan is received and verified, the data query reports that the data is unavailable and returns the anticipated time of arrival, the plan is scheduled based on the returned anticipated time of data arrival.

Test Case 12 — On-demand Processing - Within a Given Time or Proceed Anyway - Resource Unavailable

This test case verifies that when an on-demand process request is received, for a resource which is unavailable and with a specification of "process within a given time or proceed anyway", that the system schedules the request based on the anticipated availability of the resource. The plan is created to process on a resource which is unavailable during the specified time period but is scheduled to be available at some time outside of the time period specified. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the resource is unavailable during the specified time and returns the anticipated availability of the resource. The plan is scheduled based on the anticipated availability of the resource. This test is repeated for resources defined to process data types for TRMM (Rel A) (TRMM, CERES, and LIS).

Input:	On-demand processing request (with a process within a given time or proceed anyway parameter) for unavailable resource, local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data, resource, and algorithm availability, plan scheduled based on anticipated availability of the resource.
Expected Results:	The plan is received and verified, the data query confirms that the resource is unavailable during the time period specified and returns the anticipated time that the resource will be available. The plan is scheduled based on the anticipated window of opportunity.

Test Case 13 On-demand Processing - Within a Given Time or Proceed Anyway - Algorithm Unavailable

This test case verifies that when an on-demand process request is received, for an algorithm which is unavailable and with a specification of "process within a given time or proceed anyway", that the system schedules the plan based on the anticipated availability of the algorithm. The plan is created to process with an algorithm which is unavailable during the specified time period. The request parameters are verified and a data query is sent to the data server to inquire about data, resource, and algorithm availability. The data server responds that the algorithm is unavailable during the requested time period and returns the anticipated time that the algorithm will be available. The plan is then scheduled based on the anticipated window of opportunity. This test is repeated for algorithms identified to process each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or proceed anyway parameter) for unavailable algorithms, local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data, resource, and algorithm availability, plan scheduled based on the algorithms anticipated availability.

Expected Results: The plan is received and verified, the data query confirms that the algorithm is unavailable. The anticipated availability is returned and the plan is scheduled based on the algorithms window of availability.

Test Case 14 On-demand Processing - Within a Given Time or Proceed Anyway - Invalid Time Period

This test case verifies that when an on-demand process request is received, for an invalid time period and with a specification of "process within a given time or proceed anyway", that the system rejects the request. The plan is created with an invalid time period. The request parameters are verified and the invalid time period is detected. The plan is rejected based on the invalid time period. This test is repeated for each defined data type for TRMM (Rel A) (TRMM, CERES, and LIS).

Input: On-demand processing request (with a process within a given time or proceed anyway parameter) with an invalid time period.

Output: Invalid time period is detected, alarm message generated, plan rejected.

Expected Results: The plan is received and verified, the invalid time period is detected and the plan is rejected.

Test Case 15 Processing Request Report

This test case verifies that all processing requests can be verified in a report. The tester enters the command to generate the processing request report. All processing requests entered in this sequence are verified.

Input: Report command, processing requests from this sequence.

Output: Processing request report.

Expected Results: The report shows all processing requests awaiting execution.

5.1.11.2.9 Sequence 9 - On-Demand Status Requests

This series of tests verifies that when a user submits an on-demand status request, in regards to previously submitted plans, that the system responds to the request and displays the status of the requested plan. Local, remote, valid and invalid requests are demonstrated.

Test Case 1 Valid Local On-Demand Status Request

This test case verifies that after a user submits a plan to execute at their local site, that an on-demand status request can be submitted and executed. The user first submits a plan to be scheduled at their local site. After the plan is scheduled, the tester submits on-demand status requests before the plan executes, while the plan is executing, and after execution has completed. The system responds with the various status's of the plan.

Input: Processing plan to execute in the near future, on-demand status request messages.

Output: Status of plan before execution, during execution, and after execution.

Expected Results: The system responds to on-demand status requests as they are received. The system states the current state of the processing plan when the request is received. The system responds "plan in queue awaiting execution" to the first request, "plan executing" to the second request, and "plan completed" to the final request.

Test Case 2 Invalid Local On-Demand Status Request

This test case verifies that an invalid on-demand status request can be detected. The user first submits an on-demand status request for a plan that is not in the system. The system responds that the plan can not be found.

Input: On-demand status request message for a non-existent plan.

Output: Detection of invalid plan.

Expected Results: The system detects the error in the on-demand request and responds that the plan can not be found. No status is reported.

Test Case 3 Valid Remote On-Demand Status Request

This test case verifies that after a user submits a plan to execute at a remote site, that an on-demand status request can be submitted and executed. The user first submits a plan to be scheduled at a remote site. After the plan is scheduled, the tester submits on-demand status requests before the plan executes, while the plan is executing, and after execution has completed. The system responds with the various status's of the plan.

Input: Processing plan to execute in the near future at a remote site, on-demand status request messages.

Output: Status of plan before execution, during execution, and after execution.

Expected Results: The system responds to on-demand status requests as they are received. The system states the current state of the processing plan when the request is received. The system responds "plan in queue awaiting execution" to the first request, "plan executing" to the second request, and "plan completed" to the final request.

Test Case 4 Invalid Remote On-Demand Status Request

This test case verifies that a remote on-demand status for a plan not scheduled/executing at a remote site, is detected. A user submits an on-demand status request to a remote site for a non-existent plan. The remote system responds that the request was received and no information about the plan was found.

Input: Remote on-demand status request message for a non-existent plan.

Output: Detection of invalid on-demand status request.

Expected Results: The system responds to the remote on-demand status requests by declaring that no status could be found at that site for the requested plan.

5.1.11.2.10 Sequence 10 - Reprocessing Requests

This series of tests verifies that the system accepts and schedules requests for reprocessing. The scheduling for reprocessing is unique in that the system already knows the algorithms and data are available for processing. Valid and invalid requests are tested.

Test Case 1 Valid Reprocessing Request

This test case verifies that a user can submit a reprocessing plan request and that the request is validated and scheduled. The tester sends a valid reprocessing request. The system verifies that the request is valid and schedules the plan.

Input: Valid reprocessing request plan.

Output: Validation and scheduling of the reprocessing request.

Expected Results: The system responds to the request by validating and scheduling the reprocessing request.

Test Case 2 Invalid Reprocessing Request

This test case verifies that a user can submit an invalid reprocessing plan request and that the system detects the invalidity and rejects the request. The tester sends a request for reprocessing for data which has not been processed. The system responds to the request by rejecting the plan.

Input: Invalid reprocessing request plan.

Output: Detection of invalid request, plan rejection.

Expected Results: The system responds to the request by stating that the data has not been processed and is not available for reprocessing. The plan is rejected.

5.1.11.2.11 Sequence 11- Product Ordering

This series of tests verifies that after a plan has completed execution, that users can order the resulting products. Both local and remote orders are tested as well as valid and invalid requests. This test is repeated for each site available for TRMM (Rel A).

Test Case 1 Valid Local Product Ordering

This test case verifies that after a user submits a plan to execute at their local site, and the plan is executed, that the user can order the resulting products. The tester first submits a plan to be scheduled at their local site. After the plan executes, the tester submits an order for the resulting product. The order is validated and processed with the tester receiving the product.

Input: Processing plan to execute in the near future, product order request.

Output: Returned product.

Expected Results: The system validates and processes the product order. The tester receives the product generated by the plan he scheduled.

Test Case 2 Invalid Local Product Ordering

This test case verifies that an invalid product request is detected by the system. The tester submits an order for a product which has not been produced. The system responds that the product has not been generated yet. The order is rejected.

Input: Product order for a non-existing product.

Output: Detection of non-existing product, order rejection.

Expected Results: The system detects the error in the product order and responds that the product is unavailable. The order is rejected.

Test Case 3 Valid Remote Product Order

This test case verifies that after a user submits a plan to execute at a remote site, that a product order can be submitted and executed. The user first submits a plan to be

scheduled at a remote site. After the plan is executed, the tester orders the product. The system responds by verifying the request and sending the newly created product.

Input:	Processing plan to execute in the near future at a remote site, product order request.
Output:	Newly generated product.
Expected Results:	The system schedules the plan and generates the product. The ordered product is delivered to the requester.

Test Case 4 Invalid Remote Product Order

This test case verifies that an invalid remote product request is detected by the system. The tester submits a remote order for a product which has not been produced. The system responds that the product has not been generated yet. The order is rejected.

Input:	Product order for a remote non-existing product.
Output:	Detection of non-existing product, order rejection.
Expected Results:	The system detects the error in the product order and responds that the product is unavailable. The order is rejected.

5.1.11.2.12 Sequence 12 - Processing Queue Manipulation

This series of tests verifies that the processing queue can be manipulated. The test begins with a populated queue (sequence 6 - Test Case 15 - Production Queue Loading). The following tests then activate, suspend, resume, modify, and cancel the plans in their various states. A completion notice is verified when a product has been generated.

Test Case 1 Activating a Plan

This test case will display the production queue and activate a plan. The plan's status changes from scheduled to active.

Input:	Production queue, activate a plan request.
Output:	Plan status changed from scheduled to active.
Expected Results:	The plan status should change from scheduled to active.

Test Case 2 Suspending a Plan

This test case will display the production queue and suspend an active plan. The plan's status changes from active to suspended.

Input:	Production queue, suspend a plan request.
Output:	Plan status changed from active to suspend.
Expected Results:	The plan status should change from active to suspended.

Test Case 3 Resuming a Plan

This test case will display the production queue and resume a suspended plan. The plans status changes from suspended to active.

Input: Production queue, resume a plan request.

Output: Plan status changed from suspended to active.

Expected Results: The plan status should change from suspended to active.

Test Case 4 Modifying a Plan's Information

This test case will display the production queue and modify a plan's information. The information which can be modified includes priority, time, or resource allocation. The plan is modified as requested.

Input: Production queue, modify an active plan request.

Output: Plan information modified as requested.

Expected Results: The plan's information is modified as requested.

Test Case 5 Removing an Event from a Plan

This test case will display the production queue. The tester selects to modify a plan. In modifying the plan, the tester deletes an event from the plan. The plan is then updated and the tester verifies the removed event is no longer in the plan.

Input: Production queue, modify an active plan request.

Output: Plan information modified as requested.

Expected Results: The plan's information is modified as requested.

Test Case 6 Invalid Modification of a Plan

This test case will display the production queue and attempt to modify a plan's information. The information the tester attempts to modify will not be allowed. Messages indicating the attempted modification will be noted and the plan will be unchanged.

Input: Production queue, invalid modify an active plan request.

Output: Plan information should not be modified.

Expected Results: The plan's information is not modified.

Test Case 7 Canceling a Plan

This test case will display the production queue and cancel an active plan. The plan is canceled as requested.

Input: Production queue, cancel an active plan request.

Output:	Plan canceled as requested, message indicating plan was canceled is logged.
Expected Results:	The plan is canceled and a canceled plan message is logged.

Test Case 8 Completion Notice

This test case will display the production queue and allow a plan to complete execution. Once the plan completes, a completion notice is sent to the originator of the plan and the produced product is advertised as available for archive into the system.

Input:	Production queue, an active plan allowed to complete execution.
Output:	Plan Completion notification.
Expected Results:	The plan executes and a plan completion notice is sent to the plan originator. The new product is available for archive into the system.

5.1.12 Data Server Management Thread (T12-02)

5.1.12.1 Thread Objectives

The objectives of the data server management thread are:

- To provide a method to initially populate the Guide with documentation.
- To provide a method to initially populate the Directory.
- To provide a method to initially populate the Inventory.
- To provide a method to automatically update and insert metadata into the Inventory.
- To provide a method to export data server services.
- To provide a method to perform data storage, backup, and recovery of the data server/archive.
- To provide a method to distribute data via network protocols in support of a data distribution request.
- To provide a method to distribute data via an electronic media type in support of a data distribution request.

5.1.12.2 Thread Test Description

The tester will demonstrate the ability to add documentation to the guide manually using a scanner, via a file transfer, and via external media. A review of the guide is performed to determine that each documentation type added to the guide is accessible by a user. Testing is performed to verify the capability to recover from an error condition under each method of loading documentation.

Using an administrative tool, the tester demonstrates the ability to add TRMM, migrated V0, and V1 data to the guide. A review of the guide is performed to determine that each type of data added to the guide is accessible by a user.

Using an administrative tool, the tester demonstrates the ability to populate the guide with product data supported by Rel A. A review of the guide will be performed to determine that each type of data product added to the guide is accessible by a user.

Using an administrative tool, the tester demonstrates the ability to populate the directory with datasets supported by Rel A and the addition of DIF's to the directory. The directory is reviewed to determine that each dataset and/or DIF is added to the directory and is accessible by a user. Testing is performed to verify error messages are provided to the tester when invalid data is entered.

The tester demonstrates the ability to populate the inventory with TRMM data, V0 data, product data, and metadata. The automatic updating of inventory metadata is demonstrated. The inventory is reviewed to determine that each type of data is added to the inventory and is accessible by a user.

The advertising of updates to the guide, directory, and inventory services available for each data server to the advertising service is demonstrated. The tester demonstrates the capability to remotely access data added to the guide, directory, and inventory.

Testing is performed to verify proper privileges have been defined for importing and exporting of storage media as well as the system responses provided when exporting full, partially full, or non-existent storage media. Testing will verify that an export of storage media can not be performed while data is being written to the storage media. The tester will demonstrate the capability to initiate an import to a full storage media and verify the system properly imports the data to an alternate storage media and import data to a new storage media. Testing will demonstrate that an attempt to import data to a non-existent storage media will fail and that attempts to import a duplicate dataset existing on the storage media fails.

Testing will verify the capability to ingest data from all types of media (8mm tape, 4mm tape, CD ROM, etc) supported in REL A. The systems ability to respond to error conditions during ingest will be verified.

The capability to manipulate the archive inventory and directory database (to include adding, updating, and deleting entries) will be verified. Testing will verify that only those with access to a privileged account can manipulate the archive database files.

Testing is performed to verify the ability to perform data recovery of the archive and it's associated inventory database from backup media and to backup media. Testing of system response to error conditions encountered during data recovery procedure is performed. The generation of Recovery Reports is demonstrated.

The capability to support electronic and media data distributions is tested. Reformatting and compression of the data for transfer/transmission is verified. The ability to retrieve data from an offline archive is demonstrated. Proper generation of shipping and status information for media distribution requests is verified. The interface to production management for the pre-processing of data prior to data distribution is demonstrated.

Dependencies: (If Applicable)

In order to verify the objectives of the data server management thread, the following interface and functional capabilities are required:

- Top level of data server (guide, directory, and inventory) defined.
- Capability to write to and store to the guide, directory, and inventory.
- Capability to generate an advertisement of updates made to the guide, directory, and inventory.
- Access to data stored in the guide, directory, and inventory (for test verification purposes).
- Access to data and products stored on the data server.
- I/O capabilities to all types of media supported in Rel A from the data server.
- Ability to distribute data from data server in support of a distribution request.
- Interface to production management to support pre-processing of data in support of a distribution request.

Test Support Requirements

- Hardware:
  - HP735, SUN Sparc10, DEC 3000/300, IBM RS6000, SGI R3000, and Cray
- Software:
  -
- Data:
  - Documentation, Algorithms
  - Data products

5.1.12.2.1 Sequence 1 - Guide Documentation Population

The following series of tests verify the capability to populate the data server Guide with documentation. Testing of adding algorithm, core, non-core, site provider, data source, dataset, instrument, platform, product subset, project and regional area documentation to the guide are tested.

<u>Test Case 1</u> <u>Hard Copy Documentation Population</u>

Utilizing a hard copy of the associated documentation (less any algorithm documentation), the tester manually enters the data into the guide with the aid of an administrative tool.

Input: Hard copy of core, non-core, instrument, platform, product subset documentation, administrative tool to aid in entering data.

Output: Manual entry of the documentation into the guide is completed. Audit trail of entries being added to the guide is generated. Documentation is accessible via a guide search.

Expected Results: Documentation is successfully added to the guide. Review of the guide documentation shows the latest data just added.

Test Case 2 Loading Documentation Using a Scanner

This test verifies the capability to use a scanner on hard copy documentation to load the documentation into the guide.

Input: Hard copy of core, non-core, instrument, platform, product subset documentation, scanner, and administrative tool for manual updating.

Output: Hard copy of the documentation is loaded into the guide using a scanner. The documentation can be displayed for manual updates (as required) via the administrative tool prior to submission. Manual entry is completed. Audit trail of entries being added to the guide is generated. Documentation is accessible via a guide search.

Expected Results: Documentation is successfully scanned, tester is presented with option to perform manual updates prior to submission, and document is then submitted. Review of the guide documentation shows the latest data just added.

Test Case 3 Loading Documentation via External Media

This test verifies the capability to add documentation received on electronic media to the guide. This test case is repeated for each external media format supported in Rel A (i.e., CD-ROM, 8mm tape, etc).

Input: Electronic media copy of algorithm, core, non-core, instrument, platform, product subset documentation, and administrative tool for manual updating.

Output: Electronic copy of the documentation is loaded into the guide using a load media option via the administrative tool. The documentation can be displayed for manual updates (as required) via the administrative tool prior to submission. Manual entry is completed. Audit trail of entries being added to the guide is generated. Documentation is accessible via a guide search.

Expected Results: Documentation is successfully read from the external media, tester is presented with option to perform manual updates prior to submission, and document is then submitted. Review of the guide documentation shows the latest data just added.

Test Case 4 Adding Documentation Received via File Transfer

This test verifies the capability to add documentation received via electronic file transfer to the guide. This test case is repeated for each type of electronic file transfer supported in Rel A (i.e., ftp, RPC, etc).

Input: Electronic file transfer copy of algorithm, core, non-core, instrument, platform, product subset documentation, and administrative tool for manual updating.

Output: Electronic copy of the documentation is loaded into the guide using a file transfer load option via the administrative tool. The documentation can be displayed for manual updates (as required) via the administrative tool prior to submission. Manual entry is completed. Audit trail of entries being added to the guide is generated. Documentation is accessible via a guide search.

Expected Results: Documentation is successfully transferred using the file transfer option, tester is presented with option to perform manual updates prior to submission, and document is then submitted. Review of the guide documentation shows the latest data just added.

5.1.12.2.2 Sequence 2 - Errors During Guide Documentation Population

The following series of tests verify the capability to recover from errors encountered during population of the data server Guide with documentation. Testing of adding algorithm, core, non-core, site provider, data source, dataset, instrument, platform, product subset, project and regional area documentation to the guide are tested.

Test Case 1 Invalid Hard Copy Documentation Population

Utilizing a hard copy of the associated documentation (less any algorithm documentation), the tester, with the aid of an administrative tool, manually enters invalid data to populate the guide with the documentation data.

Input: Hard copy of core, non-core, instrument, platform, product subset documentation, administrative tool to aid in entering data. Tester enters invalid data for the documentation.

Output: Error messages are provided to the tester for all invalid data entered. Documentation update to the guide fails. Audit trail of entries is generated. Documentation update is rejected by the system.

Expected Results: Documentation entry rejected by the system. Error messages are provided to the tester for invalid data entry. Documentation should not be accessible on the system.

Test Case 2 Error Loading Documentation Using a Scanner

This test verifies the capability to recover from an error condition encountered while scanning hard copy documentation to load the documentation into the guide. This test case is executed for each available option on recovering from a scanning error.

Input: Hard copy of core, non-core, instrument, platform, product subset documentation, scanner, and administrative tool for manual updating. Tester produces a scanning error while scanning in the documentation.

Output: Error messages are provided to the tester when the error is encountered during the scan. Tester is allowed option to begin scanning from 'x' page, drop entire document, or start document again. Processing on document stops or resumes based on tester selected option.

Expected Results: Documentation updates to the guide depend on option selected during scan recovery. Continue from page 'x' results in the document being added to the directory. All data should exist in the guide. Dropping entire document results in the document not getting loaded into the guide. Start document again results in the first copy of the document which encountered the error not being loaded into the guide and the second (if no errors encountered) being successfully loaded.

Test Case 3 Error Loading Documentation via External Media

This test verifies the capability to recover from an error encountered when adding documentation received on electronic media to the guide. This test case is repeated for each external media format supported in Rel A (i.e., CD-ROM, 8mm tape, etc).

Input: Electronic media copy of algorithm, core, non-core, instrument, platform, product subset documentation, and administrative tool for manual updating. Tester produces a read error during the tape/disk load process.

Output: Error messages are provided to the tester when the error is encountered during the tape/disk read. Tester is allowed option to retry or abort the load process. Processing on document stops or resumes based on tester selected option.

Expected Results: Documentation updates to the guide depend on option selected during tape/disk recovery. Tester selected 'retry' results in attempt to continue to process tape/disk. If failure continues, tester has no option but to 'abort' the load attempt. If 'abort' is selected, documentation is not loaded into the guide. If 'retry' is selected and successful, documentation should be loaded into the guide.

Test Case 4 Error Adding Documentation Received via File Transfer

This test verifies the capability to recover from an error condition encountered when adding documentation received via electronic file transfer to the guide. This test case is repeated for each type of electronic file transfer supported in Rel A (i.e., ftp, RPC, etc).

Input: Electronic file transfer copy of algorithm, core, non-core, instrument, platform, product subset documentation, and administrative tool for manual updating. Tester induced error condition.

Output: Error messages are provided to the tester when the error is encountered during the file transfer. Tester is allowed option to retry or abort the transfer process. Processing on document stops or resumes based on tester selected option.

Expected Results: Documentation updates to the guide depend on option selected during file transfer recovery. Tester selected 'retry' results in attempt to continue to transfer the file. If failure continues, tester has no option but to 'abort' the file transfer attempt. If 'abort' is selected, documentation is not loaded into the guide. If 'retry' is selected and successful, documentation should be loaded into the guide.

5.1.12.2.3 Sequence 3 - Guide Data Population

The following series of tests verify the capability to populate the data server Guide with V1 data, migrated V0 data and TRMM data.

Test Case 1 V1 Data Population

This test verifies that a user can add V1 data to the Guide. This test case is executed for each type of Version 1 data that is supported in Rel A.

Input: V1 Data, administrative tool to aid in adding the V1 data to the guide.

Output: V1 data is contained in the Guide and can be accessed by an authorized user. Audit trail of entries being added to the guide.

Expected Results: V1 data was successfully added to the Guide. Review of the data in the Guide should show that the V1 data was successfully added to the guide.

Test Case 2 Migrated V0 Data Population

This test verifies that a user can add migrated V0 data to the Guide. This test case is executed for each type of Version 0 data being migrated in support of Rel A. (i.e., TOMS, CZCS, AVHRR, ERBE, ISCCP, SAGE II, SSM/I, TOVS, and SMMR).

Input: V0 migrated Data, administrative tool to aid in adding the V0 migrated data to the guide.

Output: V0 data is contained in the Guide and can be accessed by an authorized user. Audit trail of entries being added to the guide.

Expected Results: V0 data was successfully added to the Guide. Review of the data in the Guide should show that the V0 data was successfully added to the guide.

Test Case 3 TRMM Data Population

This test verifies that a user can add TRMM data to the Guide. This test case is executed for each type of TRMM data that is supported in Rel A.

Input: TRMM Data, administrative tool to aid in adding the TRMM data to the guide.

Output: TRMM data is contained in the Guide and can be accessed by an authorized user. Audit trail of entries being added to the guide.

Expected Results: TRMM data was successfully added to the Guide. Review of the data in the Guide should show that the TRMM data was successfully added to the guide.

5.1.12.2.4 Sequence 4 - Guide Product and Instrument Population

The following series of tests verify the capability to populate the guide on each data server with the required supporting product and instrument data for that server. These test cases are executed for each data product/instrument supported in Rel A.

Test Case 1 Add Product Data to Guide

This test verifies the capability to add product data to the Guide.

Input: Product data, administrative tool to aid in adding product data to the guide.

Output: Product data is contained in the Guide and can be accessed by an authorized user. Audit trail of entries being added to the guide.

Expected Results: Product data was successfully added to the Guide. Review of the data in the guide should show that the product data was successfully added to the guide.

Test Case 2 Add Instrument Data to Guide

This test verifies the capability to add instrument data to the Guide.

Input: Instrument data, administrative tool to aid in adding instrument data to the guide.

Output: Instrument data is contained in the Guide and can be accessed by an authorized user. Audit trail of entries being added to the guide.

Expected Results: Instrument data was successfully added to the Guide. Review of data in the guide should show that the instrument data was successfully added to the guide.

5.1.12.2.5 Sequence 5 - Adding Datasets to the Directory

The following series of tests verify the capability to add dataset information to the directory.

Test Case 1 Add Collection Type Dataset to Directory

This test verifies the capability to add 'collection type' datasets to the directory.

Input: Collection type dataset, administrative tool.

Output: Collection type dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Collection type was successfully added to the directory. Review of the directory should show that the collection type dataset was successfully added to the directory.

Test Case 2 Add Horizontal Coverage Code Dataset to Directory

This test verifies the capability to add horizontal coverage code datasets to the directory.

Input: Horizontal coverage code dataset, administrative tool.

Output: Horizontal coverage code dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Horizontal coverage code was successfully added to the directory. Review of the directory should show that the horizontal coverage code dataset was successfully added to the directory.

Test Case 3 Add Vertical Coverage Code Dataset to Directory

This test verifies the capability to add vertical coverage code datasets to the directory.

Input: Vertical coverage code dataset, administrative tool.

Output: Vertical coverage code dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Vertical coverage code was successfully added to the directory. Review of the directory should show that the vertical coverage code dataset was successfully added to the directory.

Test Case 4 Add Horizontal Coverage Region Dataset to Directory

This test verifies the capability to add horizontal coverage region datasets to the directory.

Input: Horizontal coverage region dataset, administrative tool.

Output: Horizontal coverage region dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Horizontal coverage region was successfully added to the directory. Review of the directory should show that the horizontal coverage region dataset was successfully added to the directory.

Test Case 5 Add Data Source Dataset to Directory

This test verifies the capability to add data source to the directory.

Input: Dataset of data source, administrative tool.

Output: Data source dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Data source was successfully added to the directory. Review of the directory should show that the data source dataset was successfully added to the directory.

Test Case 6 Add Data Type Dataset to Directory

This test verifies the capability to add data type to the directory.

Input: Dataset of data type, administrative tool.

Output: Data type dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Data type was successfully added to the directory. Review of the directory should show that the data type dataset was successfully added to the directory.

Test Case 7 Add Dataset Description to Directory

This test verifies the capability to add dataset description to the directory.

Input: Definition of a dataset description, administrative tool.

Output: Dataset description is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Dataset description was successfully added to the directory. Review of the directory should show that the dataset description was successfully added to the directory.

Test Case 8 Add Dataset Name to Directory

This test verifies the capability to add dataset names to the directory.

Input: Definition of a dataset name, administrative tool.

Output: Dataset name is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Dataset name was successfully added to the directory. Review of the directory should show that the dataset name was successfully added to the directory.

Test Case 9 Add Dataset Name Aliases to Directory

This test verifies the capability to add dataset name aliases to the directory.

Input: Definition of an alias name for a dataset, administrative tool.

Output: Dataset name alias is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Name alias was successfully added to the directory. Review of the directory should show that the name alias was successfully added to the directory.

Test Case 10 Add Dataset Name Descriptions to Directory

This test verifies the capability to add dataset name descriptions to the directory.

Input: Definition of name descriptions for a dataset, administrative tool.

Output: Dataset name description is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Name description was successfully added to the directory. Review of the directory should show that the name description was successfully added to the directory.

Test Case 11 Add Instrument Short Name to Directory

This test verifies the capability to add instrument shortname descriptions to the directory.

Input: Definition of an instrument short name. Administrative tool.

Output: Instrument short name is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Instrument shortname was successfully added to the directory. Review of the directory should show that the shortname description was successfully added to the directory.

Test Case 12 Add Platform Name to Directory

This test verifies the capability to add platform names to the directory.

Input: Definition of a platform name, administrative tool.

Output: Platform name is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Platform description was successfully added to the directory. Review of the directory should show that the platform description was successfully added to the directory.

Test Case 13 Add Product Type to Directory

This test verifies the capability to add product types to the directory.

Input: Definition of a product type, administrative tool.

Output: Product type dataset is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Product type description was successfully added to the directory. Review of the directory should show that the product type description was successfully added to the directory.

Test Case 14 Add Station ID to Directory

This test verifies the capability to add station identifiers to the directory.

Input: Definition of a station identifier, administrative tool.

Output: Station ID is contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Station ID description was successfully added to the directory. Review of the directory should show that the Station ID description was successfully added to the directory.

Test Case 15 Add Spatial and Temporal Datasets to Directory

This test verifies the capability to add spatial and temporal datasets of various resolutions to the directory.

Input: Vertical and horizontal spatial resolution datasets and temporal resolution datasets, administrative tool.

Output: Horizontal spatial resolution and temporal resolution datasets are contained in the directory and can be accessed by an authorized user. Audit trail of entries being added to the directory.

Expected Results: Spatial and temporal datasets were successfully added to the directory. Review of the directory should show that the spatial and temporal datasets were successfully added to the directory.

5.1.12.2.6 Sequence 6 - Adding Invalid Datasets to the Directory

This test case verifies that the system will not allow population of the directory with invalid datasets. This test case verifies that the system rejects entries with invalid collection types, data types, dataset descriptions, dataset names/aliases, platform names, product type, and station ID.

Test Case 1 — Enter Invalid Directory Dataset Parameters

This test verifies that the system provides the tester with an error message when invalid dataset information for the directory is entered.

Input: Known invalid dataset parameters, administrative tool.

Output: Data entry to the directory should fail. Tester receives a warning message indicating data parameters entered are invalid for the directory. No updates posted to the directory.

Expected Results: Directory update fails. Tester notified of invalid entry.

5.1.12.2.7 Sequence 7 - Initial TRMM/V0 Inventory Population

The following series of tests verify the capability to perform initial TRMM and V0 Inventory Population. These test cases are performed for each type of TRMM and V0 data supported in Rel A.

Test Case 1 — Add TRMM Data to Inventory

This test verifies the capability to add TRMM data to the Inventory. This test case is executed for each type of TRMM data supported in Rel A.

Input: TRMM data, administrative tool.

Output: TRMM data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: TRMM data was successfully added to the Inventory. Review of the Inventory shows that the TRMM data was successfully added and to the inventory.

Test Case 2 — Add V0 Data to Inventory

This test verifies the capability to add V0 data to the Inventory. This test case is executed for each type of V0 data supported in Rel A.

Input: V0 data, administrative tool.

Output: V0 data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: V0 data was successfully added to the Inventory. Review of the Inventory shows that the V0 data was successfully added and to the inventory.

5.1.12.2.8 Sequence 8 - Adding Products to the Inventory

The following series of tests verify the capability to add various types of products to the inventory. Product data is automatically updated in the inventory as a result products being produced by system processing. For testing purposes, an administrative tool is available to aid in the determination of valid inventory updates of product data.

Test Case 1 Browse Products

This test verifies the capability to automatically update the inventory with browse products.

Input: A browse product, administrative tool used to access and verify data update.

Output: Browse Product is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Browse product was successfully added to the Inventory. Review of the Inventory shows that the product was successfully added to the inventory.

Test Case 2 Cloud Coverage Products

This test verifies the capability to automatically update the inventory with cloud coverage products.

Input: Cloud coverage products, administrative tool used to access and verify data update.

Output: Cloud coverage data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Cloud coverage products were successfully added to the Inventory. Review of the Inventory shows that the product was successfully added to the inventory.

Test Case 3 Collection Type Products

This test verifies the capability to automatically update the inventory with Collection type products.

Input: Collection type products, administrative tool used to access and verify data update.

Output: Collection type data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Collection type product was successfully added to the Inventory. Review of the Inventory shows that the product was successfully added to the inventory.

Test Case 4 Coverage Code Products

This test verifies the capability to automatically update the inventory with coverage code products.

Input: Horizontal and Vertical coverage code products, administrative tool used to access and verify data update.

Output: Coverage code data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Coverage code products were successfully added to the Inventory. Review of the Inventory shows that the product was successfully added to the inventory.

Test Case 5 Coverage Region Products

This test verifies the capability to automatically update the inventory with coverage region products.

Input: Horizontal and Vertical coverage region products, administrative tool used to access and verify data update.

Output: Coverage region data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Coverage region products were successfully added to the Inventory. Review of the Inventory shows that the product was successfully added to the inventory.

Test Case 6 Data Source Products

This test verifies the capability to automatically update the inventory with data source products.

Input: Data source products, administrative tool used to access and verify data update.

Output: Data source products are contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Data source products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 7 Platform Products

This test verifies the capability to automatically update the inventory with platform products.

Input: Platform product data, administrative tool used to access and verify data update.

Output: Platform product data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Platform products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 8 Defining Product Types

This test verifies the capability to automatically update the inventory with product type definition.

Input: Product type data, administrative tool used to access and verify data update.

Output: Product type data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Product type was successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 9 Pixel Classification Products

This test verifies the capability to automatically update the inventory with pixel classification products.

Input: Pixel classification products, administrative tool used to access and verify data update.

Output: Pixel classification data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Pixel classification products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 10 QA Statistical Products

This test verifies the capability to automatically update the inventory with QA statistical products.

Input: QA statistical products, administrative tool used to access and verify data update.

Output: QA statistical product data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: QA statistical products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 11 Spatial Coverage Products

This test verifies the capability to automatically update the inventory with spatial coverage products.

Input: Horizontal and Vertical spatial coverage products, administrative tool used to access and verify data update.

Output: Spatial coverage data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Spatial coverage products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 12 Spatial Resolution Products

This test verifies the capability to automatically update the inventory with spatial resolution products.

Input: Horizontal and Vertical spatial resolution products, administrative tool used to access and verify data update.

Output: Spatial resolution data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Spatial resolution products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 13 Defining Station ID

This test verifies the capability to automatically update the inventory with a station identifier.

Input: Valid station identifier for a specified product, administrative tool used to access and verify data update.

Output: Station ID data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Station identifier was successfully added to the Inventory. Review of the Inventory shows that the ID was successfully added and to the inventory.

Test Case 14 Temporal Resolution Products

This test verifies the capability to automatically update the inventory with temporal resolution products.

Input: Temporal resolution products, administrative tool used to access and verify data update.

Output: Temporal resolution data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Temporal resolution products were successfully added to the Inventory. Review of the Inventory shows that the data was successfully added to the inventory.

Test Case 15 Defining Product Levels in the Inventory

This test verifies the capability to automatically update the inventory with granule level product data.

Input: Granule level of product data. Granule data for cloud coverage percentage, horizontal and vertical region and code coverage, granule identifiers, horizontal and vertical spatial coverage, and horizontal and vertical spatial resolution.

Output: Granule level data is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Granule level product data was successfully added to the Inventory. Review of the Inventory shows that all the data was successfully added to the inventory.

5.1.12.2.9 Sequence 9 - Auto Insert of New Metadata

The following tests verifies the capability to automatically insert new metadata into the inventory. This test case is executed for various types of metadata supported in Rel A.

Test Case 1 Metadata Insert

This test verifies the capability to automatically insert metadata into the inventory.

Input: Metadata, administrative tool used to access and verify data update.

Output: Metadata is contained in the inventory and can be accessed by an authorized user. Audit trail of entries being added to the inventory.

Expected Results: Metadata was successfully inserted to the Inventory. Review of the Inventory shows that the metadata was successfully added to the inventory.

5.1.12.2.10 Sequence 10 - Export Data Server Services

The following series of tests verify that the data server schema is properly exported when updates have been made to the guide, directory, or inventory.

Test Case 1 Export Services after Guide Update

This test verifies the exporting of the data server services after updates/changes have been made to the guide.

Input: Updates to guide information, administrative tool.

Output: Guide is updated with new information. Data server service information is updated; an advertisement is generated and exported to the advertising service. Administrator receives notification that the advertising service received the updated service announcement. Audit trail of entries being added to the guide and the notification to the advertising service. New guide data is accessible by authorized users.

Expected Results: Guide contains updated information. Advertising service received notification of new services available on the data server.

Test Case 2 Export Services after Directory Update

This test verifies the exporting of the data server services after updates/changes have been made to the directory.

Input: Updates to directory information, administrative tool.

Output: Directory is updated with new information. Data server service information is updated; an advertisement is generated and exported to the advertising service. Administrator receives notification that the advertising service received the updated service announcement. Audit trail of entries being added to the directory and the notification to the advertising service. New directory data is accessible by authorized users.

Expected Results: Directory contains updated information. Advertising service received notification of new services available on the data server.

Test Case 3 Export Services after Inventory Update

This test verifies the exporting of the data server services after updates/changes have been made to the inventory.

Input: Updates to inventory information, administrative tool.

Output: Inventory is updated with new information. Data server service information is updated; an advertisement is generated and exported to the advertising service. Administrator receives notification that the advertising service received the updated service announcement. Audit trail of entries being added to the inventory and the notification to the advertising service. New inventory data is accessible by authorized users.

Expected Results: Inventory contains updated information. Advertising service received notification of new services available on the data server.

5.1.12.2.11 Sequence 11 - Performing Exports from A Privileged Account

The following series of tests verify the operation of the export storage media function from a privileged account. Testing to verify that only users of a privileged account can perform exports to storage media.

Test Case 1 Export Full Storage Media

This test verifies the ability to export a full storage media from a privileged account.

Input: Export command from a privileged account for an on-line, full storage media.

Output: Status message, inventory report and history log.

Expected Results: The export function is executed. The status message states that the storage media has been exported. The inventory report indicates that the exported data is off-line. The history log contains an entry for the storage media export.

Test Case 2 Export Partially Full Storage - Continue/Cancel

This test verifies the system provides a response to the initiation of an export of a partially full storage media from a privileged account. The purpose of this test is to verify that the user has the option to continue or cancel the export function. This test case is executed to verify that the system responds properly to both responses (cancel and continue).

Input: Export command from a privileged account for an on-line, partially full storage media.

Output: Status message providing tester with option to cancel or continue the export of the storage media.

Expected Results: Cancel response results in export termination. Continue response results in export of storage data to specified media. Events recorded in the history log.

Test Case 3 Export Storage Media While Receiving Data

This test verifies the system provides a response to the initiation of an export of a storage media, from a privileged account, while it is receiving data.

Input: Export command from a privileged account. Ingest data to on-line storage media.

Output: Status message and error log.

Expected Results: Tester receives a status message stating that the storage media cannot be exported at this time because it is receiving data. The error log contains an entry for the failed storage media export.

Test Case 4 Export Non-existent Storage Media

This test verifies the system provides a response to the initiation of an export of a non existent storage media from a privileged account.

Input: Export command from a privileged account for a non existent storage media.

Output: Status message and error log.

Expected Results: Tester receives a status message stating that the storage media selected for export is non existent. The error log contains an entry for the failed storage media export.

Test Case 5 Export Storage Media Privilege

This test verifies that a user without the proper privileges cannot export data to storage media.

Input: Export command from a non privileged account.

Output: Status message and error log.

Expected Results: The export function is not executed. The status message states that the user under this account does not have the privilege to export storage media. The error log contains an entry for the failed storage media export.

5.1.12.2.12 Sequence 12 - Performing Imports From A Privileged Account

The following series of tests verify the operation of the import storage media function from a privileged account. Testing is performed to verify that only users with a privileged account can import data.

Test Case 1 Import To Full Storage Media

This test verifies that the system will automatically redirect the import of data to an alternate storage media if the storage media selected by the user is full.

Input: Import command from a privileged account and a full storage media.

Output: Status message, on-line storage media report, inventory report and history log.

Expected Results: Tester attempts to import data to a full storage media. System automatically imports data to an alternate storage media with available space. Tester provided with a status message indicating storage media selected is full and closed for writing; continue will import data to 'x' media. The inventory report indicates that the imported data are on-line. The history log contains an entry for the storage media import.

Test Case 2 Import to New Storage Media

This test verifies the ability to import a new storage media from a privileged account.

Input: Import command from a privileged account and new storage media.

Output: Status message, on-line storage media report and history log.

Expected Results: The import function is executed. The status message states that the storage media has been imported. The on-line storage media report indicates that the storage media is open for writing. The history log contains an entry for the storage media import.

Test Case 3 Import To Non-existent Storage Media

This test verifies the response to the import of a non existent storage media.

Input: Import command from a privileged account and a definition of a non-existent storage media.

Output: Status message and error log.

Expected Results: The import function is not executed. The status message states that the storage media selected for import is non existent. The error log contains an entry for the failed storage media import.

Test Case 4 Import of Duplicate Dataset

This test verifies the system does not allow an import of duplicate data to the storage media.

Input: Import command from a privileged account of a known dataset that already exists on the storage media.

Output: Status message and error log.

Expected Results: The import function is not executed. Tester receives a status message stating that the data selected for import already exits on the storage media. The error log contains an entry for the failed storage media import.

Test Case 5 Import Storage Media Privilege

This test verifies that a user without the proper privileges cannot import data from storage media.

Input: Import command from a non privileged account.

Output: Status message and error log.

Expected Results: The import function is not executed. The status message states that the user under this account does not have the privilege to import storage media. The error log contains an entry for the failed storage media import.

5.1.12.2.13 Sequence 13 - Ingest From Media

The following tests will demonstrate and verify the system's ability to successfully ingest data from all media types for TRMM (Rel A). Media types, for all sites to be verified are: 4 mm tape, 8 mm tape, 3480/3490 tape, CD ROM and at least one document to verify hard media ingest.

Test Case 1 Validation of Tape Media

This test case will be used to verify that it is possible to ingest valid data using electronic tape media in support of TRMM (Rel A). This test case is executed for each tape media available in Rel A.

Input: Valid ingest data passed via tape media.

Output: Data ingest from tape is successful. Data exists in the staging area. Transaction is recorded in the event log.

Expected Results: Data is successfully ingested from tape type media.

Test Case 2 Validation of CD ROM Media Type

This test case will be used to verify that it is possible to ingest valid data from CD ROM media type.

Input: Valid ingest data passed via CD ROM media.

Output: Data ingest from CD ROM is successful. Data exists in the staging area. Transaction recorded in the event log.

Expected Results: Data is successfully ingested from CD ROM type media.

Test Case 3 Validation of Hardcopy text Document media type

This test case will be used to verify that it is possible to ingest valid Hard copy text documents. The text document is scanned with a digital scanning device to produce data files and the data files are then placed in the staging area.

Input: Valid ingest data received on hardcopy text document media type. Document is scanned and produces metadata files. Metadata files are then written to the staging area.

Output: Metadata based on the scanned documents exists in the staging area. Transaction is recorded in the event log.

Expected Results: Scanning hardcopy document to produce metadata files is successful.

Test Case 4 Validation of Softcopy Text Document Media Type

This test case will be used to verify that it is possible to ingest data in softcopy format via a network.

Input: Valid ingest document/research paper received in softcopy format.

Output: The softcopy data is ingested, metadata files (if required) are generated, data stored in the staging area, and the transaction has been recorded in the event log.

Expected Results: Softcopy data ingest is successful.

5.1.12.2.14 Sequence 14 - Errors Ingesting From Media

These test's will verify the system's ability to successfully handle ingest error conditions. Error conditions include: improper mounting of ingest tape, unreadable media, incorrect operator commands, and erroneous or incomplete data on the hard media. Error conditions are displayed on the terminal in the form of status messages. These status messages are examined for complete and correct content. Any errors encountered are kept in an error log file. Data reports and data logs are examined to verify complete and accurate content.

Test Case 1 Invalid Data Format Ingest

This test case will be used to verify that the system is able to respond to an attempting to ingest a tape that contains an unexpected data format.

Input: Invalid data type on tape media.

Output: Status message is presented to the tester warning that the media mounted can not be ingested because it is an unknown datatype.

Expected Results: Software will successfully handle the ingest error condition by writing to the error log and passing error/status messages back to the user attempting to ingest the media tape.

Test Case 2 Tape Failure During Ingest

This test case will be used to verify that when a tape failure occurs during ingest, the software can correctly handle the situation.

Input: Valid ingest data passed via tape media containing data that will produce an I/O error condition during the tape read.

Output: Status message is presented to the tester warning that an I/O error has occurred during the ingest attempt.

Expected Results: Software will successfully handle the ingest error situation by writing to the error log and passing error/status messages back to the user.

Test Case 3 Validation of Hardcopy Document Media Type

This test case will be used to verify that when attempting the hard media ingest of a document, the system can handle error situations such as erroneous or incomplete data.

Input: Ingest document containing erroneous or incomplete data.

Output: Error status messages and error log entries are made.

Expected Results: Software will successfully handle the ingest error situation by writing to the error log and passing error/status messages back to the user.

Test Case 4 Ingest with Unexpected Error Conditions

This test case verifies that when unexpected error conditions occur, the system generates error messages for the user and writes the errors to the error log.

Input: Ingest data with unexpected EOF condition (incomplete dataset).

Output: Error messages passed to operator reflecting error condition.

Expected Results: Software will successfully handle the ingest error situation by writing errors to the error log and passing error/status messages back to the user.

Test Case 5 Incorrect Operator Commands

This test case will be used to verify that the system can successfully handle operator induced error conditions that may occur when ingesting data from the different media types.

Input: Data from all media types with operator errors introduced.

Output: Error status messages and error log entries are made.

Expected Results: Software will successfully handle the ingest error situation by writing errors to the error log and passing error/status messages back to the user.

5.1.12.2.15 Sequence 15 - Archive Manipulation Privileges

The following series of tests verify that only the 'administrator' of the archive can manipulate the data and the inventory of the data in the archive.

Test Case 1 Archive Command Privileges - Non-administrative

This test verifies that a user under a non-administrative account can not manipulate the data or the inventory of the data in the archive.

Input: Logon to the system under a non-administrative login.

Output: Administrative tools to manipulate the archive are not resident on the desktop. History log contains an entry for the user logon.

Expected Results: Non-administrative user does not have access to the administrative tools to manipulate the archive.

Test Case 2 Archive Command Privileges - Administrative

This test verifies that a user under an administrative account can manipulate the data and the inventory of the data in the archive.

Input: Logon to the system using a pre-defined administrative login.

Output: Administrative tools to manipulate the archive are resident on the desktop. History log contains an entry for the logon under an administrative login.

Expected Results: Administrative user has access to the administrative tools required to manipulate the archive.

5.1.12.2.16 Sequence 16 - Administrative Archive Manipulation

The following series of tests verify the ability to manipulate the data in the archive from an administrative account and subsequently have the correct information automatically maintained in the archive inventory database.

Test Case 1 Add New Data to Archive

This test verifies the ability to add new data into the archive from an administrative account and subsequently have the pertaining information automatically added into the inventory database.

Input: Archive data add command from an administrative account. Archive directory and inventory reports before data add command execution.

Output: Archive directory and inventory report after test execution and administration log. Difference files for directory and inventory database comparisons before and after test execution.

Expected Results: The archive data command is executed. The directory report after the archive data add contains the correct filenames and locations of the added data. The inventory report after the archive data add contains the correct information pertaining to the added data. The administration log contains an entry for the addition of data to the archive.

Test Case 2 Update Archive Data

This test verifies the ability to update data in the archive from an administrative account and subsequently have the pertaining information automatically updated in the inventory database.

Input: Archive data update command from an administrative account. Directory reports and inventory reports before data update command execution.

Output: Archive directory and inventory report after test execution and administration log. Difference files for directory and inventory database comparisons before and after test execution.

Expected Results: The archive update command is executed. The directory report after the archive data update contains the correct filenames and locations of the updated data. The inventory report after the archive data update contains the correct information pertaining to the updated data. The administration log contains an entry for the update to the data on the archive.

Test Case 3 Delete Archive Data

This test verifies the ability to delete data from the archive from an administrative account and subsequently have the pertaining information deleted from the inventory database.

Input: Archive data delete command from an administrative account. Directory and inventory reports before data delete command execution.

Output: Archive directory and inventory report after test execution and administration log. Difference files for directory and inventory database comparisons before and after test execution.

Expected Results: The archive delete command is executed. The directory report after the archive data delete does not contain the deleted data. The inventory report after the archive data delete does not contain the information pertaining to the deleted data. The administration log contains an entry for the deletion of data from the archive.

5.1.12.2.17 Sequence 17 - Archive Inventory Backup

The following test verifies the ability for an administrator to perform media backup saves of the archive database. Testing is performed to verify ability to recover from error conditions during backup execution.

Test Case 1 Media Backup of Archive Database

This test verifies the ability to perform a backup save of the archive database to external media. Saves are made from the administrator account.

Input: Logon to the system using a known administrator login id. Select administrative tool for archive. Select option to save the archive database information to external media.

Output: Archive database information is written to external media. Administration log contains an entry for the login and save of the database to external media.

Expected Results: Information saved to external media is the same as the data online in the archive database.

Test Case 2 Media Backup of Archive Database - Error Condition

This test verifies the ability to recover from an error condition when performing a backup save of the archive database to external media. Saves are made from the administrator account.

Input: Logon to the system using a known administrator login id. Select administrative tool for archive. Select option to save the archive database information to external media. Induce an error condition during the media backup (i.e., power down external media device).

Output: Administrator receives a warning message for the error condition with an option to continue, retry, or abort the backup save. Administration log contains an entry for the login and the error condition encountered during the save of the database to external media.

Expected Results: Database is not saved to external media. Administrator receives warning message for the error condition. System correctly handles the error condition.

5.1.12.2.18 Sequence 18 - Archive Data Backup

The following series of tests verify the ability for an administrator to perform an initial and subsequent media backup saves of the data in the archive. Due to the large amount of data on the archive, it is anticipated that a backup of the archive is performed based on a time period and that an entire archive backup would not be performed (i.e., an initial backup is performed of the entire archive at a specified time period - when the administrator performs a subsequent backup, only new data written to the archive since the last backup is saved during all subsequent backup executions). Testing is performed to verify ability to recover from error conditions during backup execution.

Test Case 1 Initial Media Backup of Archive Data

This test verifies the ability to perform an initial backup save of the data in the archive to external media. Saves are made from the administrator account.

Input: Logon to the system using a known administrator login id. Select administrative tool for archive. Select option to save archive data to external media.

Output: Entire current archive and the corresponding archive inventory database are written to external media. Administration log contains an entry for login and media backup of the archive data and database.

Expected Results: Entire archive and archive inventory database are written to external media.

Test Case 2 Subsequent Media Backup of Archive Data

This test verifies the ability to perform subsequent backup saves of the data in the archive to external media. Saves are made from the administrator account. This test case is executed only after additional data has been written to the archive.

Input: Logon to the system using a known administrator login id. Select administrative tool for archive. Select option to save archive data to external media.

Output: Only data written to the archive since the previous archive backup is written to external media. Administration log contains an entry for login and time period of media backup of the archive data and database.

Expected Results: External media save contains the inventory data base and only the data written to the archive since the previous save.

Test Case 3 Media Backup of Archive Data - Error Condition

This test verifies the ability to recover from an error condition when performing a backup save of the archive data to external media. Saves are made from the administrator account.

Input: Logon to the system using a known administrator login id. Select administrative tool for archive. Select option to save the archive data to external media. Induce an error condition during the media backup (i.e., power down external media device).

Output: Administrator receives a warning message for the error condition with an option to continue, retry, or abort the backup save. Administration log contains an entry for the login and the error condition encountered during the save of the data to external media.

Expected Results: Data is not saved to external media. Administrator receives warning message for the error condition. System correctly handles the error condition.

5.1.12.2.19 Sequence 19 - Archive Inventory Database Recovery

The following series of tests verify the ability to recover the information in the archive inventory database from an administrative account in the event of data base loss.

Test Case 1 Media Recovery

This test verifies the ability for an administrator to restore a backup media save of the archive data base in the event of data loss.

Input: Backup media save of the archive database, administrative account and tools.

Output: Media is restored. Archive data base recovered. Administration log contains an entry for login and media restore.

Expected Results: Data base information is recovered from media. Users are able to access the archive inventory data base.

Test Case 2 Media Recovery - Error Condition

This test verifies the ability to recover from an error condition encountered when restoring a media backup save of the archive database.

Input: Backup media save of the archive database, administrative account and tools. Induce an error condition during the restore (i.e., power external media source off).

Output: Media is not restored. Administrator is presented with warning message to continue, retry, or abort the media restoration procedure. Administration log contains an entry for login and failed media restore.

Expected Results: Recovery fails. System properly responds to an error condition.

Test Case 3 Manual Entry Recovery

This test verifies the ability to recover data in the archive inventory database from an administrative account manually using an administrative tool. Manual data base recovery is only used in the event of single or minimal data entry loss (i.e., one or two bad entries - not the entire database).

Input: Administrative account and tool. Data to be corrected.

Output: Entries are corrected. Administration log contains an entry for login and data corrections performed.

Expected Results: Administrator is able to manually correct bad entries in the archive inventory data base. Users are able to access the previously un-accessible inventory data.

5.1.12.2.20 Sequence 20 - Archive Data Recovery

The following series of tests verify that an administrator is able to recover archive data.

Test Case 1 Data Recovery

This test verifies that an administrator can recover data from backup media. Administrator must be aware of the dataset to be recovered and the time period it was initially saved to external media. Administration logs of data backup will contain information to aid in data recovery.

Input: Administrator account, administration logs, and backup media for data to be recovered.

Output: Data and inventory database information recovered. Administration log contains an entry for the restoration of data and database from external media.

Expected Results: The data and the inventory database is recovered. Analysis of the recovered data verifies that the inventory database and the

associated data recovered have the correct file pointers, etc. to allow access to the data.

Test Case 2 — Error Condition During Media Recovery

This test verifies the ability to handle errors that occur during recovery from backup media.

Input: Backup media of data and archive inventory database. Administrator account. Operator induced error condition (tape unit power off, etc).

Output: Status message and error log.

Expected Results: Recovery fails. The status message states that the recovery cannot be completed at this time due to the error condition encountered. The error log contains an entry for the failed data recovery. System correctly responds to the error condition.

5.1.12.2.21 Sequence 21 - Recovery Reports

The following test verifies the capability to generate reports for any data recovery performed by an administrator. Report contents should include both successful and failed data recovery execution.

Test Case 1 — Report Generation

This test case verifies the system's ability to produce a report that reflects any attempt to recover data for the archive and the inventory database from external media or manually.

Input: Administrator tool to support generation of report, administrative account. Execution of both successful and non-successful data recovery procedures.

Output: Administration log contains an entry for the execution of the report generation and the successful generation of the report.

Expected Results: Report is generated. Report contains statistics for all attempts to recover data (successful and non-successful/via media or manually).

5.1.12.2.22 Sequence 22 - Data/Product Retrieval For Electronic Distribution

The following series of tests verify the ability to retrieve products and data of all types and classes and perform any data reformatting or compression for distribution as stated in the electronic distribution request. Verification of ability to cancel an electronic distribution request is also tested. Testing will verify that distributions are handled based on the entries in the standing and retrospective order distribution list. All information related to supporting a distribution request (i.e., transmission problems, internal corrective action taken, network performance) is written to a transmission log. Transmission logs will be reviewed for supporting data content.

Test Case 1 Data Retrieval - Types

This test verifies the ability to prepare the various types of data supported in Rel A for electronic distribution. This test case is executed for all the data types that are supported in Rel A (i.e., instrument, ancillary, correlated, etc).

Input: Distribution request for a specific type of data.

Output: Status messages and distribution request logs. Electronic transfer of data. Transmission log contains entry for request supported.

Expected Results: The status messages state that the data has been retrieved. The distribution request logs contain entries for the electronic distribution request. Analysis of the data show that the retrieved data conforms to the information in the electronic distribution request.

Test Case 2 Data Retrieval - Classes

This test verifies the ability to prepare the various classes of data supported in Rel A for electronic distribution. This test case is executed for all the data classes that are supported in Rel A (i.e., L0 instrument, L1 instrument, etc).

Input: Distribution request for a specific class of data.

Output: Status messages and distribution request logs. Electronic transfer of data. Transmission log contains entry for request supported.

Expected Results: The status messages state that the data has been retrieved. The distribution request logs contain entries for the electronic distribution request. Analysis of the data show that the retrieved data conforms to the information in the electronic distribution request.

Test Case 3 Data Retrieval - Products

This test verifies the ability to prepare the various product data supported in Rel A for electronic distribution. This test case is executed for all the data products that are supported in Rel A (i.e., higher level product orders, metadata, QA data, etc).

Input: Distribution request for a specific data product.

Output: Status messages and distribution request logs. Electronic transfer of data. Transmission log contains entry for request supported.

Expected Results: The status messages state that the data product has been retrieved. The distribution request logs contain entries for the electronic distribution request. Analysis of the data show that the retrieved data conforms to the information in the electronic distribution request.

Test Case 4 Data retrieval (off-line data)

This test verifies the ability to prepare off-line data for electronic distribution.

Input: Distribution request for a specific set of data from the offline archive.

Output: Status messages and distribution request logs. Electronic transfer of data. Transmission log contains entry for request supported.

Expected Results: The status messages state that the data product has been retrieved. The distribution request logs contain entries for the electronic distribution request. Analysis of the data show that the retrieved data conforms to the information in the electronic distribution request.

Test Case 5 Data retrieval (non existent data)

This test verifies the system provides a valid response to an electronic distribution request for non existent data.

Input: Distribution request for a known set of non-existent data.

Output: Status messages and error log.

Expected Results: The status message states that the data in the electronic distribution request is non existent. The error log contains an entry for an electronic distribution request for non existent data. The request is not processed.

Test Case 6 Reformatting Data for Distribution

This test verifies the ability to reformat data for an electronic distribution. This test case is executed for each type of data that can be reformatted for distribution.

Input: Distribution request requiring data be reformatted prior to data distribution.

Output: Status message and distribution request log. Data reformatted and electronically distributed. Transmission log contains entry for request supported.

Expected Results: Data located, reformatted, and distributed. Distribution log contains an entry for the electronic distribution and the status of the request after successful distribution.

Test Case 7 Compressing Data for Distribution

This test verifies the ability to reformat data for an electronic distribution. This test case is executed for each type of data that can be reformatted for distribution.

Input: Distribution request requiring data be compressed prior to data distribution.

Output: Status message and distribution request log. Data compressed and electronically distributed. Transmission log contains entry for request supported.

Expected Results: Data located, compressed, and distributed. Distribution log contains an entry for the electronic distribution and the status of the request after successful distribution.

Test Case 8 Interface to Production Processing

This test verifies the ability to interface with the production management thread. A distribution request requiring pre-processing prior to data distribution requires an interface to production planning and processing to schedule the processing activities. The data server management thread will locate the data and provide notification to the production planning to schedule the processing. Production planning will schedule and perform the processing and provide a notification to data server management of processing completion. Data server management will then distribute the data.

Input: Distribution request requiring pre-processing prior to data distribution.

Output: Notification to production planning for data processing requirement and location of data to be processed. Data processed. Receipt of notification from production planning on processing completion. Successful distribution of processed data. Status messages, event log, distribution log, and transmission log.

Expected Results: Data located, production processing notified, processing performed, and notification of data availability for distribution upon processing completion received. Data distributed.

Test Case 9 Distribution request cancellation

This test verifies the ability to cancel an electronic distribution request. An initial distribution request is received and is being processed by the system. A cancellation message to cancel the request for the data distribution is received.

Input: Distribution request for data. Cancellation of distribution request.

Output: Status messages and distribution request log.

Expected Results: Initial request being processed. Receipt and processing of the cancellation message results in the termination of the retrieval of the data for distribution. Any allocated disk space, etc. is relinquished.

Test Case 10 Distribution request status report

This test verifies the capability to generate a distribution request status report for electronic distribution requests received during a given time period. The status report will include information such as the priority of the distribution request processed, the distribution directions, and other data related to processing the order.

Input: Administrative tool to aid in the generation of the report. Distribution log files.

Output: Distribution request status report.

Expected Results: The distribution request status report contains the correct information concerning the actions taken on each distribution request handled by the system during the specified report time period.

Test Case 11 Manual Override

This test verifies the capability to perform a manual override to alter the priority of a distribution request.

Input: Distribution request. Interactive tool to allow operator to manually override (modify) the priority of a pending distribution request.

Output: Priority of distribution request status in the queue changed based on manually assigned priority (higher or lower in the queue). Queue log contains an entry for the change in the priority of the distribution request. Originator of request receives notification of new anticipated time of completion for the distribution request.

Expected Results: Pending distribution moved within the queue to new anticipated time of completion. Originator notified of change in estimated time of data availability.

Test Case 12 Data Protection

This test verifies there are multiple levels of data protection established based on the sensitivity levels of ECS data in accordance with NHB 2410.9

Input: Several distribution requests by both authorized and non-authorized users. Distribution requests will require access to highly sensitive data (restricted access). Both authorized and non-authorized users will request access to the restricted data.

Output: Data authorization verified prior to data distribution. Non-authorized users are not granted access to restricted data. Distribution requests to non-authorized users for highly sensitive data is denied. Distribution log contains an entry for each request received, verification of authorization, and distribution disposition (granted, denied, issued, etc).

Expected Results: Highly sensitive data is not released to un-authorized users.

5.1.12.2.23 Sequence 23 - Data Retrieval For Media Distribution

The following series of tests verify the ability to retrieve data and write it to all types of media supported in Rel. A for distribution. Verification of the proper generation of shipping and status information is performed. Verification of ability to cancel a media distribution request is also tested.

Test Case 1 Data retrieval - All Media Types

This test verifies the capability to write data to all types of in support of a media distribution request. This test case is executed for each media type supported in Rel A (i.e., CD ROM, 8mm, etc).

Input: Distribution request for data to be received on a specific type of media.

Output: Status messages, distribution logs, and physical media. Data transferred to physical media requested.

Expected Results: The status messages state that the data has been retrieved. The distribution request logs contain entries for the media distribution requests. Analysis of the data show that the retrieved data conforms to the information in the media distribution requests.

Test Case 2 Data retrieval (off-line data)

This test verifies the ability to prepare off-line data for media distribution.

Input: Distribution request for a specific set of data from the offline archive.

Output: Status messages and distribution request logs. Data transferred to media for distribution.

Expected Results: The status messages state that the data product has been retrieved. The distribution request logs contain entries for the media distribution request. Analysis of the data show that the retrieved data conforms to the information in the media distribution request.

Test Case 3 Bit Error Rate

This test verifies the response to a bit error rate that is higher than the set threshold for a media distribution request.

Input: Distribution request for data to be received on a particular type of media.

Output: Status messages, distribution logs, and physical media. Initial data transfer to media fails due to excessive bit error rate. Successive media mount and transfer of data is successful.

Expected Results: The status messages request the mounting of new media due to a higher than allowable bit error rate. Successive attempt successful. The distribution request logs contain entries for the distribution request. Analysis of the data show that the retrieved data conforms to the information in the distribution request.

Test Case 4 Reformatting Data for Distribution

This test verifies the ability to reformat data for a media distribution. This test case is executed for each type of data that can be reformatted for distribution.

Input:	Distribution request requiring data be reformatted prior to data distribution.
Output:	Status message and distribution request log. Data reformatted and written to media for distributed.
Expected Results:	Data located, reformatted, and distributed to media. Distribution log contains an entry for the electronic distribution and the status of the request after successful distribution.

Test Case 5 Compressing Data for Distribution

This test verifies the ability to reformat data for an electronic distribution. This test case is executed for each type of data that can be reformatted for distribution.

Input:	Distribution request requiring data be compressed prior to data distribution.
Output:	Status message and distribution request log. Data compressed and electronically distributed.
Expected Results:	Data located, compressed, and distributed. Distribution log contains an entry for the electronic distribution and the status of the request after successful distribution.

Test Case 6 Interface to Production Processing

This test verifies the ability to interface with the production management thread. A distribution request requiring pre-processing prior to data distribution requires an interface to production planning and processing to schedule the processing activities. The data server management thread will locate the data and provide notification to the production planning to schedule the processing. Production planning will schedule and perform the processing and provide a notification to data server management of processing completion. Data server management will then write the data to the media specified in the distribution request.

Input:	Distribution request requiring pre-processing prior to data distribution.
Output:	Notification to production planning for data processing requirement and location of data to be processed. Data processed. Receipt of notification from production planning on processing completion. Data successfully written to media for distribution. Status messages, event log, and distribution log.
Expected Results:	Data located, production processing notified, processing performed, and notification of data availability for distribution upon processing completion received. Data written to media for distribution.

Test Case 7 Distribution request cancellation

This test verifies the ability to cancel a media distribution request. An initial distribution request is received and is being processed by the system. A cancellation message to cancel the request for the data distribution is received.

Input: Distribution request for data. Cancellation of distribution request.

Output: Status messages and distribution request log.

Expected Results: Initial request being processed. Receipt and processing of the cancellation message results in the termination of the retrieval of the data for distribution. Any allocated disk space, etc. is relinquished.

Test Case 8 Distribution request status report

This test verifies the capability to generate a distribution request status report for media distribution requests received during a given time period. The status report will include information such as the priority of the distribution request processed, the distribution directions, and other data related to processing the order.

Input: Administrative tool to aid in the generation of the report. Distribution log files.

Output: Distribution request status report.

Expected Results: The distribution request status report contains the correct information concerning the actions taken on each distribution request handled by the system during the specified report time period.

5.1.12.2.24 Sequence 24 - Shipping And Handling Information

The following test verifies the ability to generate the correct shipping and handling information for any distribution request requiring the data be distributed on media vice electronically.

Test Case 1 Generate Shipping and Handling Information

This test verifies the generation of the correct shipping and handling information for a media distribution request.

Input: Media distribution request. Administrative tool to aid in the generation of the shipping information.

Output: Shipping label, packing slip, and information to aid the user in accessing the media/utilizing the data upon receipt.

Expected Results: The shipping label and packing slip contain the correct information as stated in the media distribution request. The information provided to aid the user is informative and can aid a novice user.

Test Case 2 Modify Shipping and Handling Information

This test verifies the ability to modify shipping and handling information for a media distribution request.

Input: Media distribution request. Administrative tool to aid in the generation of the shipping information. Generate shipping and handling information with incorrect data (i.e., incorrect address). Re-call account data with incorrect shipping data and correct invalid data.

Output: Corrected shipping label, packing slip, and information to aid the user in accessing the media/utilizing the data upon receipt. Administration log contains an entry for the generation of the shipping label and the subsequent modification to the shipping data.

Expected Results: Shipping labels can be modified without requiring re-generation of all the data on the label. Modified label and packing slip contain the correct information as stated in the media distribution request. The information provided to aid the user is informative and can aid a novice user.

5.1.12.2.25 Sequence 25 - Supporting Product Generation

The following series of tests verify the interface between data ingest, the data server, and product generation. Testing will verify that disk space is allocated to support the generation of product data.

Test Case 1 Generating Product Data

This test verifies that disk space is properly allocated to support the generation of product data.

Input: Notification from production planning of data requiring allocation of disk space to support generation of product data. Notification shall contain anticipated amount of disk space to be allocated.

Output: Disk space allocated to support generation of product data. Event log contains an entry for the allocation of disk space to support data processing.

Expected Results: Disk space allocated. Processing completes.

5.1.12.2.26 Sequence 26 - Supporting Data Archival

The following series of tests verify the capability to archive the various levels of data received. Data archived is saved for long term storage. Test cases are executed for each applicable data source interface supported as part of TRMM (Rel A) (i.e., EDOS, EOS, etc).

Test Case 1 Archiving Production Data Sets

This test verifies the capability to take production data sets from temporary storage and write them to permanent archive. Testing will verify that the corresponding directory and inventory information is updated.

Input: Production data sets ingested and saved to temporary storage. Data available for archival message from ingest service.

Output: Data written to permanent archive. Event log contains entry for the write of data from temporary storage to permanent archive. Notification to data ingest of successful data archival. Generation of data availability notice.

Expected Results: Data written to permanent archive. Disk space allocated for temporary storage relinquished. Directory and inventory information has been updated. User is able to access the data.

Test Case 2 Archiving Standard Products

This test verifies the capability to take standard products from temporary storage and write them to permanent archive. Testing will verify that the corresponding directory and inventory information is updated.

Input: Standard product data ingested and saved to temporary storage. Data available for archival message from ingest service.

Output: Data written to permanent archive. Event log contains entry for the write of data from temporary storage to permanent archive. Notification to data ingest of successful data archival. Generation of data availability notice.

Expected Results: Data written to permanent archive. Disk space allocated for temporary storage relinquished. Directory and inventory information has been updated. User is able to access the data.

Test Case 3 Archiving TRMM Data

This test verifies the capability to take TRMM data from temporary storage and write them to permanent archive. This test case is executed to verify that each type of TRMM data ingested can be stored to permanent archive and the corresponding directory/inventory information is updated.

Input: TRMM data ingested and saved to temporary storage. Data available for archival message from ingest service.

Output: Data written to permanent archive. Event log contains entry for the write of data from temporary storage to permanent archive. Notification to data ingest of successful data archival. Generation of data availability notice.

Expected Results: Data written to permanent archive. Disk space allocated for temporary storage relinquished. Directory and inventory information has been updated. User is able to access the data.

Test Case 4 Archiving V0 Data

This test verifies the capability to take V0 data from temporary storage and write them to permanent archive. This test case is executed to verify that each type of V0 data ingested can be stored to permanent archive and the corresponding directory/inventory information is updated.

Input: V0 data ingested and saved to temporary storage. Data available for archival message from ingest service.

Output: Data written to permanent archive. Event log contains entry for the write of data from temporary storage to permanent archive. Notification to data ingest of successful data archival. Generation of data availability notice.

Expected Results: Data written to permanent archive. Disk space allocated for temporary storage relinquished. Directory and inventory information has been updated. User is able to access the data.

Test Case 5 Manually Mounting Archive Media

This test verifies the capability to manually mount archive media in the event of an automated system failure.

Input: System failure resulting in failure to automatically mount archive media. Operator interactive tool to support manual mounting of archive media.

Output: Operator able to manually mount archive media. Data in staging area written to permanent archive. Inventory and directory information correctly updated. Data advertisement generated.

Expected Results: Manual mounting of archive media successful. Staged data written to permanent archive. Disk space allocated for staged data relinquished. Inventory and directory updated. Advertisement generated. Users able to access data.

Test Case 6 Bit Error Corrections

This test verifies the capability to support archival data integrity. Bit error corrections for data written to the archive shall not exceed 1 in 10 to the 12th.

Input: Ingested data requiring bit error corrections.

Output: Data written to the staging area. Ingest verification of data requires bit error corrections be performed on the data prior to data archival. Bit error corrections shall not exceed 1 in 10 to the 12th. Higher bit error corrections result in data validation failure and no data archival.

Expected Results: If bit error conditions are less than 1 in 10 to the 12th, data archival to permanent storage is successful. Higher bit error conditions result in data validation failure and notification issued to originator for data validation failure. Temporary staging disk space de-allocated.

5.1.12.2.27 Sequence 27 - Supporting Status Requests/Issuing Status Messages

The following series of tests verify the ability to generate status messages for distribution and data staging requests. Status messages can be generated in response to a query or automatically as a result of satisfying a request.

Test Case 1 Distribution Status Request

This test verifies the capability to generate a status message in response to a query of the status of a pending or completed distribution request. Testing will verify the ability to acknowledge the query and issue a response to both an outstanding/pending request and a fulfilled request.

Input: Query for status of a distribution request.

Output: Response to query. Distribution log contains an entry for receipt of the query and generation of the response.

Expected Results: Response for query is generated.

Test Case 2 Automatic Generation of Distribution Status

This test verifies the capability to automatically generate a status message upon completion (fulfillment) of a distribution request. Status messages will automatically be issued to the IMS.

Input: Distribution request.

Output: Data/Product distributed to requester. Status message automatically generated and issued to the IMS notifying IMS of the disposition of the distribution request. Distribution log contains an entry for the data distribution and generation of the status message.

Expected Results: Status message automatically generated to IMS for completion of a distribution request.

Test Case 3 Staging Status Requests

This test verifies the capability to generate staging status messages for requests for retrieval of data and/or products.

Input: Staging status query for outstanding distribution request.

Output: Staging status message issued to requester. Distribution log contains an entry for the receipt of the staging status message and the generation of the response.

Expected Results: Staging status message issued to requester.

5.1.13 System Integration Build 3 (B12)

The System Integration Build 3 is an integrated build of the following system threads and builds:

- System Integration Build 1
- Product Generation Thread
- Data Server Management Thread

5.1.13.1 Build Objectives

The objectives of the system integration build 3 are:

- Provide system access capabilities, management, and communication services, and system security.
- Provide the capability to ingest TRMM ancillary data and perform preprocessing of ancillary data.
- Provide the capability to perform production planning, scheduling, and product generation.
- Provide the capability to support on-demand processing requests.
- Provide the capability to store, advertise, and distribute data/products to/from the data server. Functionality includes automatic updating of the top level of the data server (directory and inventory data) and distribution of data/products electronically via network communications and physical media.
- Provide the Quality Assurance (QA) interface for metadata and products.

5.1.13.2 Build Test Description

Testing is performed to verify system access capabilities and system accountability, data product generation, on-demand processing, and the QA interface for metadata and products.

Dependencies: (If Applicable)

In order to verify the objectives of the system integration build 3, the following interface and functional capabilities are required:

- Interface to ingest service to support ancillary data ingest into the system.
- Interface to production generation system to support product planning, scheduling, and generation.
- Interface to data server to support product management (archiving of data products, exporting of data).

Test Support Requirements

- Hardware:
  –
- Software:
  –
- Data:
  –

5.1.13.2.1 Sequence 1 - System Accessibility and Accountability

The following series of tests verify the ability for a registered and non-registered user to logon to the system, verify the functionality of communication services, system security, and management services.

Test Case 1 Registered User Logon - Accessibility and Accountability

This test verifies that a registered user is able to logon to the system and obtain a particular data set or product. Testing will verify user authorization to obtain the data set or product. The data set or product requested will be transferred to the user via ftp to verify communication services. All transactions will be logged in the event log and reviewed/printed to verify the functionality of the management services.

Input:	Logon to the ECS system as a known registered user via the scientist workbench. Locate a data set or product that is accessible to the login id used and transfer the data via ftp to the local workbench.
Output:	Data set or product is transferred to the local scientist workbench and can be displayed by the user. Login and transfer of data to the local workbench is recorded in the event log.
Expected Results:	Registered user able to log directly onto the ECS system without being prompted to complete a registration form. User access the data set or product is verified prior to the data being transferred to the local workstation. Data is transferred via ftp to verify communication services. Event log is printed to verify completeness of management services (system accountability) of transactions being performed on the system.

Test Case 2 Non-Registered User Logon - Accessibility and Accountability

This test verifies that a non-registered user is able to logon to the system, register, and obtain a particular data set or product. Testing will verify user authorization to obtain the data set or product. The data set or product requested will be transferred to the user via RPC to verify communication services. All transactions will be logged in the event log and reviewed/printed to verify the functionality of the management services.

Input:	Logon to ECS system as a known non-registered user, complete the registration process, and download the scientist workbench to the local workstation. Bring up the scientist workbench and logon to the ECS system. Locate a data set or product that is accessible to the login id used and transfer the data via RPC to the local workbench.
Output:	Data set or product is transferred to the local scientist workbench and can be displayed by the user. Login, registration, and transfer of data to the local workbench is recorded in the event log.
Expected Results:	Non-registered user able to log onto the ECS system, complete the registration process, and download the workbench to the local workstation.

Test Case 3 Multiple User - Accessibility and Accountability

This test verifies that correct logistical data is collected when multiple users are accessing the system simultaneously. Variations of accessibility to the system are performed by

each user (i.e., data search, data request, data transfer). Verification of each users accessibility to data (if the user requested data or is attempting to transfer data) is performed. All transactions performed by each user is logged in the event log. The event log is printed to verify that management services is logging the correct transactions for each user.

Input: Multiple registered and non-registered users logon to the ECS system. Non-registered users complete the registration process. All users perform a variety of tasks (i.e., data search, data request, data transfer, registration, browsing of data).

Output: Non-registered users complete registration. All users (previously registered and non-registered) obtain search results, data products, etc. based on function performed. Login and activities performed by each user are recorded in the event log. Any data files transferred are logged in the transaction log.

Expected Results: User(s) access the data set or product is verified prior to the data being transferred to the local workstation. Data is transferred via ftp, RPC, or other electronic means to verify communication services. Event log is printed to verify completeness of management services (system accountability) of multiple transactions being performed.

5.1.13.2.2 Sequence 2 - Data Product Generation

This test verifies that the entire data product generation sequence is operational. The test sequence begins with TRMM ancillary data being preprocessed and ingested into the ECS system. A production plan is then created and loaded into the production queue. Then the job is executed and the product is distributed. Data ordering procedures are verified as well as data backup availability. Machine failure is examined. Reprocessing plans are entered into the system and executed.

Test Case 1 TRMM Ancillary Preprocessing and Ingest

This test case verifies that when data is received into the system that the system recognizes the data type and performs preprocessing prior to ingesting the data into the system. This test case uses each type of TRMM ancillary data which requires preprocessing. Attempts to archive multiple versions of selected data is performed. Archives of current versions of data is made, allowing the preceding version to be deleted. A data availability notice is sent to the production planning system upon completion of the ingest.

Input: Each type of TRMM ancillary data which requires preprocessing.

Output: Data availability notice, archive inventory reflects addition of new data to the data server. Ingest log contains entries for each data ingest.

Expected Results: Data type is determined by the ingest service, preprocessing is performed, and the data is ingested into the system. A data

availability notice is sent to production planning process for each data ingest. A data availability notice is sent to the production planning system, for each data ingest, to advertise that the data is available for product generation.

Test Case 2 Process Plan Creation/Queue Loading

This test case verifies that a valid request for creating a processing plan is acted upon by the system. The operator enters a create a plan message. The message must contain a start time, an end time, and planning criteria. Planning criteria must contain a processing algorithm. The system verifies the parameters and generates the plan. Periodically during this queue loading process, status reports are generated. A plan is generated to include each data type ingested in test case 1. The plans are loaded into the production queue.

Input: Create a plan message with valid start and finish times, valid planning criterion, defined required resources.

Output: The plans are generated, the resources are allocated.

Expected Results: Successful plan messages are sent to the data server and the SMC, the plans and the associated metadata are sent to the data storage system.

Test Case 3 Queue Manipulation

This test case verifies that the production queue can be manipulated. As jobs are scheduled to be executed, the jobs are canceled, suspended, and resumed. Each manipulation of the queue results in display updates and resource reallocation throughout the system. As each queue manipulation is executed, the appropriate responses are verified.

Input: Queue manipulation commands, queue loaded with plans from test case 2.

Output: The system responds to each queue manipulation command, displays are updated, resources reallocated.

Expected Results: As production queues are manipulated, the system responses are verified.

Test Case 4 Completion Notice

This test case displays the production queue and allows a plan to complete execution. Once the plan completes, a completion notice is sent to the originator of the plan and the produced product is advertised as available for archive into the system.

Input: Production queue, an active plan allowed to complete execution.

Output: Plan completion notification, resource deallocated.

Expected Results: The plan executes and a plan completion notice is sent to the plan originator. The new product is available for archive into the system.

Test Case 5 Data Ordering

This test case verifies user data orders. Data can be ordered for processing or the actual products can be ordered. This test case verifies all the different data orders a user can request. All data order information is recorded in the log files.

Input: Data orders.

Output: Data distribution, log files updated.

Expected Results: The data is distributed as ordered. The log files reflect the orders.

Test Case 6 Data Backup Availability

This test case verifies that data which is destroyed can be recovered from a backup source. A plan for selected data is entered into the system. While the plan is in the queue, the requested data is marked as deleted. When the plan is executing, the missing data is discovered. Backup resources are used to recover the data and the plan continues successfully.

Input: Production plan, delete data request, data recovery request.

Output: Alarms detecting missing data, production plan completion, history log updates.

Expected Results: The deleted data is recovered from the backup sources. The plan which was suspended, is resumed and completes successfully. History logs are updated to reflect the data restoration.

Test Case 7 Machine Backup Availability

This test case verifies that when a machine being used for product generation goes down, a backup machine is available to continue the processing. A plan is entered into the system for product generation. During processing, one of the allocated resources goes down. The event is discovered, and a backup resource replaces the downed machine. Processing continues successfully.

Input: Production plan, resource schedule.

Output: Alarms detecting downed resource, production plan completion, history log updates.

Expected Results: The production plan is interrupted when the resource goes down. However, once the switch over occurs, the processing continues and the plan completes successfully. The history log is updated to reflect these events.

Test Case 8 Reprocessing Plan Generation

This test case verifies that a valid request for reprocessing data is acted upon by the system. The operator enters a reprocessing plan message. The system verifies the parameters and resources needed, and generates the plan.

Input: Create a reprocessing plan message with valid start and finish times, valid planning criterion, defined required resources.

Output: The plan is generated, the resources are allocated.

Expected Results: Successful plan messages are sent to the data server and the SMC, the plan executes successfully.

5.1.13.2.3 Sequence 3 - On Demand Processing

This series of tests verifies that on-demand processing requests for all data types are processed as received. Three types of on-demand requests are verified; 1) process as soon as possible, 2) process within a specified time, or reject, and 3) process within a given time, or proceed anyway. Only valid requests are examined in this integration build.

Test Case 1 On-demand Processing - As Soon As Possible

This test case verifies that when an on-demand process request is received with a specification of "as soon as possible", that the system verifies the request and adds the plan to the current schedule in the first available window of opportunity. The plan is created for data which is known to be in the archives. A data query is sent to the data server to inquire about data availability. The resource schedule is verified for resource availability. The requested algorithm is verified as available for processing. The plan is then incorporated into the schedule based on resource and data availability. The tester verifies that the plan is inserted into the schedule at the first available opportunity. This test is repeated for each defined data type for Release A (TRMM, CERES, and LIS).

Input: On-demand processing request (with an as soon as possible parameter), local site schedule for data, resource, and algorithm availability.

Output: Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled at first window of opportunity.

Expected Results: The plan is received and verified, the data query confirms that the data, resource, and algorithms are available, and the plan is scheduled for the first window of opportunity.

Test Case 2 On-demand Processing - Within a Given Time or Reject

This test case verifies that when an on-demand process request is received with a specification of "within a give time or reject", that the system verifies the request and adds the plan to the current schedule within the specified time. The plan is created for data which is known to be in the archives. A data query is sent to the data server to

inquire about data availability. The resource schedule is verified for resource availability during the specified time period. The requested algorithm is verified as available for processing during the requested time period. The plan is then incorporated into the schedule based on the requested time period. The tester verifies that the plan is inserted into the schedule as requested. This test is repeated for each defined data type for Release A (TRMM, CERES, and LIS). Some plans will be rejected based upon data availability and resource conflicts.

Input:	On-demand processing request (with a process within a given time or reject), local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled for requested time, plan rejection notices.
Expected Results:	The plan is received and verified, the data query confirms that the data, resource, and algorithms are available for the requested time period, and the plan is scheduled to execute during the requested time period. Plans that cannot be scheduled within the given time are rejected. Rejection notices must give reasons for rejection.

Test Case 3	On-demand Processing - Within a Given Time or Proceed Anyway

This test case verifies that when an on-demand process request is received with a specification of "within a give time or proceed anyway", that the system verifies the request and adds the plan to the current schedule within the specified time. The plan is created for data which is known to be in the archives. A data query is sent to the data server to inquire about data availability. The resource schedule is verified for resource availability during the specified time period. The requested algorithm is verified as available for processing during the requested time period. The plan is then incorporated into the schedule based on the requested time period. The tester verifies that the plan is inserted into the schedule as requested. This test is repeated for each defined data type for Release A (TRMM, CERES, and LIS).

Input:	On-demand processing request (with a process within a given time or proceed anyway), local site schedule for data, resource, and algorithm availability.
Output:	Data query to data server concerning data availability, resource availability, and algorithm availability, plan scheduled for requested time.
Expected Results:	The plan is received and verified, the data query confirms that the data, resource, and algorithms are available for the requested time period, and the plan is scheduled to execute during the requested time period.

5.1.13.2.4 Sequence 4 - Quality Assurance Process

This series of tests verifies that quality assurance processing can be performed on metadata and products. Metadata and products are given the different ratings available and reports are produced to verify that all ratings are recorder correctly. The list of products awaiting QA is displayed and updated as production QA is completed.

Test Case 1 Metadata QA Update

This test case verifies that metadata received into the system is quality reviewed. Once the metadata is generated or received into the system, the QA office is notified of pending metadata. The metadata is then quality checked and the appropriate fields in the metadata are marked. This test case will use all possible metadata ratings available. A quality report is generated and verified.

Input: Metadata awaiting QA, QA commands, report generation command.

Output: Metadata QA fields updated, quality report.

Expected Results: The QA metadata fields are updated and the quality report reflects the metadata updates.

Test Case 2 Product QA Update

This test case verifies that products generated or received into the system are quality reviewed. Once the product is generated or received into the system, the QA office is notified of the pending product as well as the expected time of QA completion. The QA monitoring display reflects the products awaiting QA and the expected time of completion. The product is then quality checked and the appropriate fields in the product metadata are marked. This test case will use all possible product ratings available. A quality report is generated and verified. The QA monitor display is updated as QA of the product is completed.

Input: Products awaiting QA, QA commands, report generation command.

Output: Product metadata QA fields updated, quality report, QA monitor display update.

Expected Results: The QA product metadata fields are updated and the quality report reflects the QA completion. The QA monitor display is updated as products are added to the list and when QA is completed for the product.

5.1.14 CSMS System Monitoring Thread (T13-01)

5.1.14.1 Thread Objectives

The objectives of the system monitoring thread are:

- Provide agent services to include physical device services, system software services, an object request broker, and DCE based security.
- Provide system monitoring functionality for data transfers, user sessions, network configuration, policy and procedures, and event monitoring.
- Provide fault management functions to include alarm processing, manual event log analysis, fault isolation, identification and recovery, and vendor diagnostics.
- Provide performance management functions to include network performance, event logging, and statistical data collection.
- Provide enhanced configuration management functionality to include baseline management, remote software distribution, and license administration.
- Provide accountability functions to include user registration audit trail and price estimation.
- Provide schedule management functions to include local network and system configuration management, network resource discovery and display, and system and site enhancement analysis.
- Provide report generation services.

5.1.14.2 Thread Test Description

Testing verifies the various objectives of the system monitoring thread. Agent service functionality is tested. All agents are verified to ensure that the services they are intended to provide are functional. Various features of DCE security are tested in a DCE environment.

System monitoring functionality is tested. External and internal data transfers are monitored. Errors are induced into the process to verify error detection during data transfers. User session monitoring is tested to verify idle session disconnect. Network configuration monitoring is tested by comparing the displays to actual site configuration. Policy and procedure monitoring is verified for accuracy. Event monitoring is tested for schedules and events.

Fault management functionality is tested. Various fault conditions are simulated to verify that alarms are generated. Event logs are analyzed for completeness and accuracy. Fault isolation, identification and recovery procedures are verified. Vendor provided diagnostics are also tested.

Testing is performed to verify that performance management tools meet the system requirements. Statistics are gathered and analyzed for network performance.

Testing verifies that the configuration management tool supports the system requirements. The capability to perform baseline configuration management of the system software is demonstrated. Remote distribution of the software under the CM tool is performed and the ability to maintain administrative control of software licenses is tested.

Testing demonstrates the ability to maintain accountability of actions performed on the system. Generation of both user registration and user audit trails are tested. Price estimation capabilities are verified.

Schedule management functions are tested for both local network and system configuration. Network resource discovery and display information is verified. System and site enhancement analysis is tested using OA tools.

Testing verifies the capability to perform report generation services. Reports for network statistics are generated and verified. All applicable COTS packages which generate reports are executed and the reports verified. Host level access reports are verified.

Dependencies: (If Applicable)

In order to verify the objectives of the system monitoring thread, the following interface and functional capabilities are required:

Interface to the agent services to verify customized, physical device, and system software services agents.

Interface to the DAACs to verify network monitoring, DCE security, and schedule management.

Interface to the HP OpenView tool to verify network and system fault management.

Interface to the statistical collection data base to insert and dump statistical information.

Test Support Requirements

- Hardware:
  - LAN Analyzer
  - DCE Configured LAN
- Software:

  Protocol Decoder
  - Network Management
  - DCE software/Security server
  - Monitoring and Logging software
- Data:

5.1.14.2.1 Sequence 1 - Agent Services

This series of tests will verify that agent services are functional. Test cases are executed to verify customized agents, physical device services and system software services. The object request broker is tested to verify that the RPC's are functional. Test cases are executed to verify DCE based security.

<u>Test Case 1</u> <u>Customized Agents</u>

This test case verifies that all customized agent services can be executed by the system. Customized agents are generated to perform specific system functions which are repeated throughout the system. Where applicable, test drivers are used to simulate the interface using the customized agent.

Input:	Test drivers for customized agents.
Output:	Reports, event logs, or any other means of determining the success of the execution of the customized agent services.
Expected Results:	The reports and event logs verify that the customized agent is executed as expected.

Test Case 2 Physical Device Services

This test case verifies that all physical device services are executable. These physical device services are generated to monitor, access, and control physical devices within the system. Test drivers are utilized where necessary to verify the device services. Reports and event logs are used to verify test results.

Input:	Test drivers for physical device services.
Output:	Reports, event logs, or any other means of determining the success of the execution of the physical device service.
Expected Results:	The reports and event logs verify that the physical device services agents are operational.

Test Case 3 System Software Services

This test case verifies that all system software services are executable. System software services are generated to perform operating system services that are used throughout the system. An example of a system software service is file transfers using ftp. Test drivers are utilized where necessary to verify the system software services. Reports and event logs are used to verify test results.

Input:	Test drivers for system software services. Network analyzer.
Output:	Reports, event logs, or any other means of determining the success of the execution of the system software services.
Expected Results:	The reports and event logs verify that the system software services agents are operational.

Test Case 4 DCE Based Security

This test case verifies that the DCE based security is operational. The tester attempts unauthorized login into the system. The DCE based security reports the unauthorized login attempts. The login attempts are recorded in the event logs.

Input:	Unauthorized logon attempts.
Output:	Alarms detecting unauthorized logon attempts, record of attempts in the event log.
Expected Results:	The system detects the unauthorized login attempts. The attempts are logged in the event log.

Test Case 5 Virus Checker

This test case verifies that viruses in the system are can be identified and captured within the system. The DCE based security reports the presence of the virus and the recommended approach to capture it.

Input: Virus simulation program.

Output: Alarms detecting the presence of the virus, recommendation of steps to capture the virus.

Expected Results: The system detects the virus and recommends the appropriate steps to eliminate it from the system.

5.1.14.2.2 Sequence 2 - System Monitoring

This test sequence verifies the system monitoring functionality. Both external and internal data transfers are monitored. Errors are induced into the data transfer process to verify error detection. User session, network configuration, policy and procedures and event logging are also verified in this test sequence.

Test Case 1 External Data Transfers

This test case verifies that data transfers between external system components can be monitored by the system. Testers initiate data transfers between external system elements and use the HP OpenView tool to monitor the success of the transfer. Log files are updated to reflect the successful transfer.

Input: HP OpenView, data transfer commands.

Output: Log file updates.

Expected Results: The system monitors the transfer between external elements. The data is transferred successfully. The log file is updated to reflect the transferred data.

Test Case 2 External Data Transfer - Errors

This test case verifies that errors encountered during external transfers are monitored by the system. Testers initiate data transfers between external system elements and use the HP OpenView tool to monitor the transfer. The transfer is interrupted by the tester. The HP OpenView tool alarms the operators of the interruption. The log file is updated to reflect the failed attempt.

Input: HP OpenView, data transfer commands, transfer interruption.

Output: Alarms, log file updates.

Expected Results: The system monitors the transfer between external elements. The data transfer is interrupted with HP OpenView discovering the failed transfer. The log file is updated to reflect the failed transfer request.

Test Case 3 Internal Data Transfers

This test case verifies that data transfers between internal system components can be monitored by the system. Testers initiate data transfers between internal system elements and use the HP OpenView tool to monitor the success of the transfer. Log files are updated to reflect the successful transfer.

Input: HP OpenView, data transfer commands.

Output: Log file updates.

Expected Results: The system monitors the transfer between internal elements. The data is transferred successfully. The log file is updated to reflect the transferred data.

Test Case 4 Internal Data Transfer - Errors

This test case verifies that errors encountered during internal transfers are monitored by the system. Testers initiate data transfers between internal system elements and use the HP OpenView tool to monitor the transfer. The transfer is interrupted by the tester. The HP OpenView tool alarms the operators of the interruption. The log file is updated to reflect the failed attempt.

Input: HP OpenView, data transfer commands, transfer interruption.

Output: Alarms, log file updates.

Expected Results: The system monitors the transfer between internal elements. The data transfer is interrupted with HP OpenView discovering the failed transfer. The log file is updated to reflect the failed transfer request.

Test Case 5 User Session Monitoring

This test case verifies that user sessions are monitored by the system monitoring tool. A tester will log into the system and then leave the session idle. At the pre-defined time, the HP OpenView tool will acknowledge that the session is idle, and relinquish the connection. The log file is updated to reflect the relinquished connection.

Input: HP OpenView, user session.

Output: Relinquished connection, log file updates.

Expected Results: The system monitors the user session and once the pre-defined time has expired for an inactive session, the session is relinquished. The log file is updated to reflect the relinquished session.

Test Case 6 Network Configuration

This test case verifies that the HP OpenView tool monitors the network configuration. Both local network configuration status, as well as all systems at the ESN network management facility are displayed using the tool. The display is verified for accuracy.

Input: HP OpenView, network configuration request.

Output: Display of local and system network configuration.

Expected Results: The HP OpenView tool reflects the current system configuration.

Test Case 7 Policies and Procedures

This test case verifies that the policies and procedures within the ECS system can be viewed and modified (as applicable). The tester displays the policies and procedures for various functions within the system. If applicable, the tester modifies entries in the tables.

Input: Policy and procedures display, modification tool.

Output: Display of policy and procedures, altered information as applicable.

Expected Results: The policy and procedures are viewed and altered as applicable.

Test Case 8 Event Monitoring

This test case verifies that the each elements schedule and events are monitored. The HP OpenView tool is used to monitor the schedules and events. The tester uses the tool to view the elements schedules and events. As events occur, the tester verifies that the displays are updated. New events added to the schedule also update the display.

Input: HP OpenView, schedules and events.

Output: HP OpenView displays.

Expected Results: As events occur, the displays are updated to reflect event completion. New events added to the schedule are reflected on the displays.

5.1.14.2.3 Sequence 3 - Fault Management

This series of test verifies that fault management processing is functional. Test cases are executed to create system faults to verify alarms are being generated and displayed. Event logs are verified to ensure all system fault alarms recorded. Network faults are created to verify fault isolation, identification and recovery processes. Vendor diagnostic tools are executed and the results analyzed.

Test Case 1 Local System Fault Detection

This test case verifies that local system faults can be detected by the fault management system. The tester induces faults into the local system. The monitoring tool, HP OpenView, responds to the faults by generating alarms and displaying the location of the fault on the system map.

Input: Tester induced local system faults, HP OpenView.

Output: Alarms detecting the fault, display update of the fault location, and event log update of fault detection.

Expected Results: The fault is detected by the fault management system and alarms are generated indicating fault detection. The display image of the system configuration indicates where the fault occurred. The event log is displayed to ensure that the fault was recorded.

Test Case 2 Remote System Fault Detection

This test case verifies that remote system faults can be detected by the fault management system. The tester induces faults into the remote systems. The monitoring tool, HP OpenView, responds to the faults by generating alarms and displaying the location of the fault on the system map. This test is repeated for each site supported in Release A.

Input: Tester induced system faults, HP OpenView.

Output: Alarms detecting the fault, display update of the fault location, and event log update of fault detection.

Expected Results: The fault is detected by the fault management system and alarms are generated indicating the fault detection. The display image of the system configuration indicates where the fault occurred. The event log is displayed to ensure that the fault was recorded.

Test Case 3 Network Fault Detection

This test case verifies that network faults can be detected by the fault management system. The tester induces network faults into the system. The monitoring tool, HP OpenView, responds to the network faults by generating alarms and displaying the location of the fault on the system map. This test is repeated for each site supported in Release A.

Input: Tester induced remote network faults.

Output: Alarms detecting the fault, display update of the fault location, and event log update of fault detection.

Expected Results: The fault is detected by the fault management system and alarms are generated indicating the fault is detected. The display image of the system configuration indicates where the fault occurred. The event log is reviewed to ensure that the fault was recorded.

Test Case 4 Recovery Procedures

This test case verifies that when network faults are detected that the procedures for recovery correct the problem. The tester induces network faults into the system. The monitoring tool, HP OpenView, responds to the network faults by generating alarms and displaying the location of the fault on the system map. The recovery procedures are then followed to ensure that the connection to the remote site can be reestablished. Fault detection records, as well as execution of recovery procedures, are recorded in the event log.

Input: Network faults at remote sites.

Output: Alarms detecting the fault, display update of the fault location, and event log update of fault detection.

Expected Results: The fault is detected by the fault management system and alarms are generated indicating the fault is detected. The display image of the system configuration indicates where the fault occurred. Recovery procedures are executed. The event log is displayed to ensure that the fault and recovery are recorded.

Test Case 5 Vendor Diagnostics

This test case verifies that diagnostic tools provided by COTS packages can be executed , the results validated, and that the tool supports the needs of the system. This test case is run on each vendor provided diagnostic test. Error conditions are induced to verify the functionality of the diagnostics. This test case is executed for each diagnostic tool provided to support Release A.

Input: Vendor provided diagnostic tools, tester induced error conditions.

Output: Error detection and diagnostic audit log.

Expected Results: The induced error is detected by the diagnostic tool and logged. The error detection logged reflects the induced error.

5.1.14.2.4 Sequence 4 - Performance Management

This series of tests verify that performance management processing is functional. Test cases are executed to validate network performance and network trending. Reports are generated to verify event logging of network activity and statistical data collection.

Test Case 1 Network Performance

This test case verifies that network performance statistics are accurately recorded and stored in the data base. Testers perform tests to generate network statistics which are recorded in the data base. The tester records the statistical data as it is generated. The tester then generates a network performance report and compares the information in the report with the information recorded during the test. Operational maintenance records are included in network performance statistics.

Input: Network performance statistic generating tests, network performance report command.

Output: Statistics recorded during test, network performance report.

Expected Results: The network performance report matches those statistics recorded by the tester during the test.

Test Case 2 Network Trending

This test case verifies that network trending statistics are accurately recorded and stored in the data base. Testers perform tests to generate network trending statistics which are recorded in the data base. The tester records the statistical data as it is generated. The

tester then generates a network trending report and compares the information in the report with the information recorded during the test.

Input:	Network trending statistic generating tests, network trending report command.
Output:	Statistics recorded during test, network trending report.
Expected Results:	The network trending report matches those statistics recorded by the tester during the test.

5.1.14.2.5 Sequence 5 - Enhanced Configuration Management

This series of test verifies that the enhanced configuration management functions are operational. A series of regression tests of IR1 configuration management services are executed first. These tests verify baseline management. Then, remote software distribution techniques are verified. and license administration is tested. Failure of the configuration management tool is tested for reliability of the backup process.

Test Case 1 — Process (Check-in) Algorithm to Configuration Management

This test verifies the ability to receive algorithms by the science community and bring them under configuration management control under the designated CM Tool, ClearCase.

Input:	Science Algorithms. Designated CM repository and CM Tool. Network environment to support file transfer or tape I/O capability.
Output:	Appropriate status/error messages and log entries generated as a result of loading science algorithm into the CM Tool.
Expected Results:	The Science Algorithm loaded should be resident under the CM Tool and accessible by all users.

Test Case 2 — Check-out Single Algorithm from Configuration Management To Multiple Users

This test verifies the ability to check out a single algorithm under CM Control to multiple users of the science community.

Input:	Designated CM repository and CM Tool, algorithms resident under CM Control, network environment and valid users that can "check-out" an algorithm from CM.
Output:	Appropriate status/error messages and log entries generated as a result of checking science algorithm out from the CM Tool.
Expected Results:	Version logged out reflects "checked out" to "xxx user". Multiple users should be able to check out the same version of an algorithm while only the first user to "check-out" the algorithm will have the "reserved' copy.

Test Case 3 Process (Check-in) Modified Algorithms to Configuration Management

This test verifies the ability to receive modified algorithms from the science community and check them into Configuration Management under the designated CM Tool, ClearCase. This test demonstrates baseline management.

Input: Original version for the modified Science Algorithm must already be resident in CM. Modified science algorithms. CM repository and CM Tool. Network environment to support file transfer or tape I/O capability.

Output: Appropriate status/error messages and log entries generated as a result of loading science algorithm into the CM Tool. Updated Version of science algorithm.

Expected Results: Original Version 1 of the Science Algorithm plus updated version (Version 1 plus differences to make up new version of the algorithm) resident under the CM Tool and accessible by all users.

Test Case 4 Remote Software Distribution

This test verifies the remote software distribution process. The tester will use the CM tool, ClearCase to build a tar file intended for software distribution. Once the file is created, the tester logs onto a remote site and gets the tar file. Installation procedures are verified at the remote site.

Input: Commands to generate a tar file for remote distribution, files needed to generate the tar file.

Output: Tar file generated and transferred.

Expected Results: The tester is able to generate the tar file and transfer it to a remote site. The tester is then able to use the installation procedures to install the software at the site.

Test Case 5 License Administration

This test verifies the system's license administration functions. The tester first installs a COTS package at a remote site. The tester then generates a license administration report which reflects the COTS package installation at that location.

Input: COTS package to install at a remote site, license agreement.

Output: License administration report.

Expected Results: The license administration report reflects the newly installed COTS package at the remote site.

Test Case 6 Configuration Management Failure

This test verifies that when the system's configuration management system fails, that the backup system can be installed and operations can continue. While using the

configuration management tool, the tester induces a fatal data base fault into the system. The severity of the fault indicates to the tool that the data base is corrupt. The backup procedures are then executed to replace the corrupted data base with the backup copy. Differences between the backup copy and the original data base are determined and appropriate procedures are taken to correct the differences.

Input: Configuration Management tool, tester induced fatal data base error, recovery procedures.

Output: Alarms indicating fatal error, updated data base installed, entire event captured in the configuration management logs.

Expected Results: The configuration management tool should be able to detect the fatal error, be updated with the backup copy, and the differences between the original and the backup copy are resolved.

5.1.14.2.6 Sequence 6 - Accountability

This series of test verifies that the systems accountability of user registration and user audit trails are valid. These tests generate multiple user registrations and verify that all information is recorded and logged. Each user account will access the system and the audit trails of time spent on the system is verified. Price estimation is verified.

Test Case 1 Multiple User Registration

This test case verifies that multiple user registrations are recorded and logged. The tester generates multiple user registrations. The tester then prints the audit trail for the users that were created. The information in the audit trail is verified.

Input: Multiple user registration information, audit trail printout.

Output: Audit trail printout.

Expected Results: The audit trail printout should contain all the user registration information entered by the tester.

Test Case 2 User Access Audit Trail

This test case verifies that each user registered in test case 1 can access the system and that actions performed by each user is recorded in an audit trail. The tester logs into each created account and performs different functions. Some accounts are used to perform identical functions. Users request data to be transferred. The audit trail report is then generated to ensure that all the user's activity is recorded correctly.

Input: Multiple user registration accounts, audit trail printout.

Output: Audit trail printout.

Expected Results: The audit trail printout should contain all the users' activity as entered by the tester.

Test Case 3 Price Estimation

This test case verifies the price estimation functions of the system. A tester logs on as a registered user to perform various functions on the system. Before performing functions which will incur a cost, the tester records the price estimation of that function. Once the function is performed, the tester compares the actual cost with the estimated cost.

Input: Functions to perform which will incur a cost to the user.

Output: Actual cost report.

Expected Results: The actual cost incurred falls within a given boundary of the estimated cost of the executed functions.

5.1.14.2.7 Sequence 7 - Schedule Management

This series of test verifies that the schedule management processes are functional. Local network management processes are tested. System configuration processes are verified. Tests are executed via the network, to verify that resource discovery and display are functional. System and site enhancement analysis using OA tools are verified.

Test Case 1 Local Network Management Processes

This test case verifies that local network management processes are functional. The tester attempts to modify the local network configuration. Both valid and invalid changes are attempted. Valid attempts are successful, invalid attempts are stopped upon system detection and the network management personnel are notified. The events are captured in the event log.

Input: Local network configuration change.

Output: Network configuration changes, alarms indicating invalid network configuration change attempt, event log updates.

Expected Results: Valid network changes are acted upon. The system responds to the invalid attempted changes and the network management personnel are notified. The event log records the incident.

Test Case 2 System Configuration Processes

This test case verifies that the system configuration processes are functional. The tester attempts to modify the system configuration. The attempt is stopped upon detection and the system configuration personnel are notified. The event is captured in the event log.

Input: System configuration change.

Output: Alarms indicating system configuration change attempt, event log updates.

Expected Results: The system responds to the attempted change and the system configuration personnel are notified. The event log records the incident.

Test Case 3 Resource Discovery and Display

This test case verifies that resources added to or taken away from the system are discovered and reported on. The tester first adds a device to the system. The network monitoring tool reflects the update to the system resources. The tester then removes the device. Again, the network monitoring tool reflects the update.

Input: Adding a device to the system, removing the device to the system.

Output: Network monitoring tool display update, network monitoring report, event log update.

Expected Results: The network monitoring tool records the added device and updates the network monitoring tool to advertise it's existence. The network monitoring report reflects the activity as well as the event log.

Test Case 4 Site Enhancement Analysis

This test case verifies that when a site configurations is enhanced, the site is not negatively affected. The tester attempts to enhance the site with an updated version of a COTS package. Site performance is not affected.

Input: Enhanced version of a COTS package, audit trail of site installation, network usage statistics.

Output: Event log of update to site, network statistics report.

Expected Results: The site's copy of the COTS package is updated and the update is recorded in the event log. The network statistics support the fact that the sites performance by enhancing the package are not affected.

Test Case 5 System Enhancement Analysis

This test case verifies that when the system configurations is enhanced, the system is not negatively affected. The tester attempts to enhance the system with an updated version of a COTS package. System performance is not affected.

Input: Enhanced version of a COTS package, audit trail of system installation, network usage statistics.

Output: Event log of update to system, network statistics report.

Expected Results: The system's copy of the COTS package is updated and the update is recorded in the event log. The network statistics support the fact that the system's performance by enhancing the package are not affected.

5.1.14.2.8 Sequence 8 - Report Generation Services

This series of test verifies that all report generation services are operational. Test cases are executed to generate reports of network statistics. These reports are validated. All reporting

capabilities associated with COTS packages are executed and the reports validated. Reports via host level access are executed and verified.

Test Case 1 Fault Reports

This test case verifies that all system faults generated during the System Monitoring Thread test are captured and reported on in the fault reports. The information within the fault reports are verified.

Input: Report generation commands.

Output: Fault reports.

Expected Results: The fault reports are generated and the information contained within the reports are verified.

Test Case 2 Performance Reports

This test case verifies that all system performance statistics generated during the System Monitoring Thread test are captured and reported on in the performance reports. The information within the performance reports are verified.

Input: Report generation commands.

Output: Performance reports.

Expected Results: The performance reports are generated and the information contained within the reports are verified.

Test Case 3 Configuration Change Reports

This test case verifies that all system configuration changes generated in the System Monitoring Thread tests are captured and reported on in the configuration change reports. The information within the configuration change reports are verified.

Input: Report generation commands.

Output: configuration change reports.

Expected Results: The configuration change reports are generated and the information contained within the reports are verified.

Test Case 4 System Access Reports

This test case verifies that all system access statistics generated in the System Monitoring Thread tests are captured and reported on in the system access reports. The information within the system access reports are verified.

Input: Report generation commands.

Output: System access reports.

Expected Results: The system access reports are generated and the information contained within the reports are verified.

Test Case 5 Accountability Reports

This test case verifies that all system accountability statistics generated during the System Monitoring Thread tests are captured and reported on in the accountability reports. The information within the accountability reports are verified.

Input: Report generation commands.

Output: Accountability reports.

Expected Results: The accountability reports are generated and the information contained within the reports are verified.

Test Case 6 Virus Detection Reports

This test case verifies that all system virus detection statistics generated in the System Monitoring Thread tests are captured and reported on in the virus detection reports. The information within the virus detection reports are verified.

Input: Report generation commands.

Output: Virus detection reports.

Expected Results: The virus detection reports are generated and the information contained within the reports are verified.

Test Case 7 Security Breach Reports

This test case verifies that all system security breach statistics generated are captured and reported on in the security reports. The information within the security reports are verified.

Input: Report generation commands.

Output: Security reports.

Expected Results: The security reports are generated and the information contained within the reports are verified.

Test Case 8 Data Receipt Reports

This test case verifies that all data received into the system is recorded in the data base. The tester ingests data into the system and generates the data receipt report to verify that the information about the data is recorded.

Input: Report generation commands, data to ingest.

Output: Data receipt reports.

Expected Results: The data receipt reports are generated and the information contained within the reports are verified.

Test Case 9 Data Access/Distribution Reports

This test case verifies that all data access attempts are recorded in the data base. The tester accesses data ingested in test case 8 and inputs distribution requests. The data access report is then generated and the tester's access attempt is verified in the report.

Input: Report generation commands, data access commands, data distribution commands.

Output: Data access reports.

Expected Results: The data access reports are generated and the information contained within the reports are verified.

Test Case 10 Network Reports

This test case verifies that all network statistics, such as network round trip delay, network reset and restart indications, outages and CRC errors, and network performance statistics are collected and included in network reports. All network activity generated during the execution of this thread test are verified in the network reports.

Input: Report generation commands, network access.

Output: Network reports.

Expected Results: The network reports are generated and the information contained within the reports are verified.

Test Case 11 User Services Reports

This test case verifies that all user sessions are recorded in the data base and an user session report can be generated. Information in the report includes user name, service identification, date/time stamp, time expended, facilities used, etc. The tester compares the report to the activities performed in the system monitoring thread.

Input: Report generation commands, user sessions from the system monitoring thread tests.

Output: User Services report.

Expected Results: The user services report is generated and the information contained within the report is verified.

Test Case 12 Training Reports

This test case verifies that all training information in the system can be obtained in a training report. The tester generates the training report and verifies that all the training information on the bulletin board is included in the report.

Input: Report generation commands, bulletin board with training information on it.

Output: Training report.

Expected Results: The training report is generated and the information contained within the report is verified against the training bulletin board.

Test Case 13 Archive Reports

This test case verifies that all data and products contained in the archive can be verified via an archive report. The report contains information about each archive and the information contained within it. The tester verifies the content of the report.

Input: Report generation commands.

Output: Archive report.

Expected Results: The archive report is generated and the information contained within the report is verified.

Test Case 14 Configuration Status Reports

This test case verifies that information about all hardware and system and scientific software in the system can be obtained. The tester generates a configuration status report. The information in the report is verified.

Input: Report generation commands, hardware site installation information, software installation information.

Output: Configuration Status report.

Expected Results: The configuration status report is generated and the information contained within the report is verified.

Test Case 15 Proposed Enhancement Reports

This test case verifies that all proposed enhancements proposed to the system can be obtained from the system. The tester generates the report and verifies the information obtained. The report contains information about proposal name, description of enhancement, rationale, impacts, costs and schedule milestones.

Input: Report generation commands.

Output: Proposed enhancement report.

Expected Results: The proposed enhancement report is generated and the information contained within the report is verified.

Test Case 16 User Recommendation Reports

This test case verifies that all user recommendations entered into the system data base can be retrieved. The tester generates the user recommendation report and verifies the information. Information in the report includes user information, type of transaction, satisfaction statistics, and user recommendations.

Input: Report generation commands.

Output: User recommendation report.

Expected Results: The user recommendation report is generated and the information contained within the report is verified.

5.1.15 User Services Thread (T13-02)

5.1.15.1 Thread Objectives

The objectives of the user services thread are as follows:

- Provide the user interface for verification of data/product level of access privileges.
- Provide the user interface for access to the data dictionary, data server, LIM, and advertising service.
- Provide the user interface for physical data/product distribution requests.
- Provide the user with a hypertext linked help function in order to obtain help for data across a heterogeneous network. Information in the help menu must support the most novice user.

5.1.15.2 Thread Test Description

Testing demonstrates the verification of user assigned data and product access level privileges. Verification that a limited access user can not obtain restricted data and products is performed. Testing to verify that the system provides the user with a status message for a query for non-existent data or products is performed.

Testing demonstrates the ability for the user to directly access the data server, data dictionary, LIM, and advertising service. Verification of user feedback to the generation of an invalid query is performed.

Testing demonstrates the ability for the user to generate distribution requests and product orders. Testing will verify that all required fields must be completed prior to the submission of the request or order. Testing to verify that the system is able to generate a response to the user to an invalid request or order is performed. Testing also verifies that the system provides the user with an acknowledgment to a distribution request or product order.

Testing is performed to determine the hypertext help functionality implemented to guide the user through the ECS system. Help data is reviewed to determine if it is useful to all levels of users.

Dependencies: (If Applicable)

- In order to verify the objectives of the user services thread, the following interface and functional capabilities are required:
- Data server with both product level data and raw data (L0, L1, etc.).
- User access directly to the data server, data dictionary, LIM, and advertising service.
- Interface to production planning to support pre-processing of user requested data prior to data distribution.
- User authorization and authentication verification.
- Communication capabilities to support data distribution (i.e., ftp, RPC, etc.) to the user.

- Communication capabilities to provide responses to user queries, requests, and orders.

Test Support Requirements

- Hardware:
  - TBS
- Software:

- Data:
  -

5.1.15.2.1 Sequence 1 - Data/Product Data Access Privileges

The following tests verify that user privileges for data access to both products and raw data were properly defined during the user registration process. Attempts to access the ECS system and perform queries as an authorized user with limited and full privileges are performed. These tests are performed for each type of data/product available on the system.

Test Case 1 Restricted Data Access - Limited Privileges

This test verifies that a user with limited privileges can not access restricted data/products.

Input: Log onto the ECS system as a known limited access user. Formulate a query for a known set of restricted data/product. Submit the query.

Output: User receives a notice that access to the data was denied because the user privileges do not support access to the data/product requested. History log contains messages of the events that have transpired.

Expected Results: The user is not allowed to access restricted data/product based on privileges defined in the user's profile. An audit record is created.

Test Case 2 Unrestricted Data Access - Limited Privileges

This test verifies that a user with limited privileges is granted access to unrestricted data/products.

Input: Log onto the ECS system as a known limited access user. Formulate a query for a known set of un-restricted data/product. Submit the query.

Output: User receives a response to the query. Access to the data/product is granted. History log should contain messages of the events that have transpired.

Expected Results: The user is allowed to access the unrestricted data/product based on privileges defined in the user's profile. An audit record is created.

Test Case 3 Restricted Data Access - Full Privileges

This test verifies that a user with full privileges is granted access to restricted data/products.

Input: Log onto the ECS system as a known privileged user. Formulate a query for a known set of restricted data/product. Submit query.

Output: User receives a response to the query. Access to the data/product is granted. History log contains messages of the events that have transpired.

Expected Results: The user is allowed to access the restricted data/product based on privileges defined in the user's profile. An audit record is created.

Test Case 4 Unrestricted Data Access - Full Privileges

This test verifies that a user with full privileges is granted access to unrestricted data/products.

Input: Log onto the ECS system as a known privileged user. Formulate a query for a known set of unrestricted data/product. Submit the query.

Output: User receives a response to the query. Access to the data/product is granted. History log contains messages of the events that have transpired.

Expected Results: The user is allowed to access the unrestricted data/product based on privileges defined in the user's profile. An audit record is created.

5.1.15.2.2 Sequence 2 - Query For Non-Existent Data/Product

The following tests verify the system provides the user with a response to a query for non-existent data or product.

Test Case 1 Non-existent Data Query

This test verifies that the user receives a response when a query for non-existent data is submitted.

Input: Log onto the ECS system and formulate a query for access to non-existent data.

Output: Query response. History log contains messages of the events that have transpired.

Expected Results: Originator receives a message indicating data does not exist. An audit record is created.

Test Case 2 Non-existent Data Product Query

This test verifies that the user receives a response when a query for a non-existent data product is submitted.

Input: Log onto the ECS system and formulate a query for access to a non-existent data product.

Output: Query response. History log contains messages of the events that have transpired.

Expected Results: Originator receives a message indicating data product does not exist. An audit record is created.

5.1.15.2.3 Sequence 3 - User Access to ECS Components

The following test cases verify the user's ability to access the data server, data dictionary, LIM, and advertising service is verified. User authorization is verified prior to access. Both expert and intermediate access are demonstrated. System responses to the user for an invalid query is verified.

Test Case 1 Accessing the Data Server

This test verifies the capability for the user to generate a query to the data server.

Input: Scientist workbench and generation of a query for a known set of data on a particular data server.

Output: Query response. User is able to browse or view data received in query response. Event log contains entries for query generation and submission to the data server and query response received.

Expected Results: Query results package received by the user.

Test Case 2 Utilizing Data Server Help

This test verifies the capability to obtain help information for the data server across a heterogeneous network.

Input: Workbench interface to a help function for the data server.

Output: User able to obtain access to help function for information pertaining to the data server. Information provided to the user is meaningful and will aid the user in accessing the various components of the data server. Event log contains an entry for the activation of the help function.

Expected Results: User able to access site help function and navigate through the help data pertaining to the data server.

Test Case 3 Accessing the Data Dictionary

This test verifies the capability for the user to generate a query to the data dictionary.

Input: Scientist workbench and generation of a query for a list of data products defined on the ECS system.

Output: Query response. User is provided list of data products. Event log contains entries for query generation and submission to the data dictionary and query response received.

Expected Results: Query results package received by the user.

Test Case 4 Utilizing Data Dictionary Help

This test verifies the capability to obtain help information for the data dictionary across a heterogeneous network.

Input: Workbench interface to a help function for the data dictionary.

Output: User able to obtain access to help function for information pertaining to the data dictionary. Information provided to the user is meaningful and will aid the user in accessing the various components of the data dictionary. Event log contains an entry for the activation of the help function.

Expected Results: User able to access site help function and navigate through the help data pertaining to the data dictionary.

Test Case 5 Accessing the LIM

This test verifies the capability for the user to generate a query to the LIM.

Input: Scientist workbench and generation of a query for a list of data products defined for a particular site.

Output: Query response. User provided with a list of data products. Event log contains entries for query generation and submission to the LIM and query response received.

Expected Results: Query results package received by the user.

Test Case 6 Utilizing Data LIM Help

This test verifies the capability to obtain help information for the LIM across a heterogeneous network.

Input: Workbench interface to a help function for the LIM.

Output: User able to obtain access to help function for information pertaining to the LIM. Information provided to the user is meaningful and will aid the user in accessing the various components of the LIM. Event log contains an entry for the activation of the help function.

Expected Results: User able to access site help function and navigate through the help data pertaining to the LIM.

Test Case 7 Accessing the Advertising Service

This test verifies the capability for the user to generate a query to the advertising service.

Input: Scientist workbench and generation of a query for a list of services available on the ECS system.

Output: Query response. User provided list of services. Event log contains entries for query generation and submission to the advertising service and query response received.

Expected Results: Query results package received by the user.

Test Case 8 Utilizing Advertising Service Help

This test verifies the capability to obtain help information for the advertising service across a heterogeneous network.

Input: Workbench interface to a help function for the advertising service.

Output: User able to obtain access to help function for information pertaining to the advertising service. Information provided to the user is meaningful and will aid the user in accessing the various components of the advertising service. Event log contains an entry for the activation of the help function.

Expected Results: User able to access site help function and navigate through the help data pertaining to the advertising service.

Test Case 9 Generation of Invalid Query

This test verifies the system provides the user with a response to an invalid query request. This test case is executed for each interface available to the user (i.e., direct data server access, LIM, etc).

Input: Scientist workbench and generation of an invalid query (i.e., missing data, not enough data for system to satisfy query, etc).

Output: Query response. User provided a response to the query informing the user of the inability to satisfy the query. Event log contains entries for query generation and query response received.

Expected Results: User receives a response to an invalid query or a query with insufficient data.

5.1.15.2.4 Sequence 4 - User Generation of Distribution Requests

The following tests verify the capability to generate complete, accurate distribution requests. Testing will also be performed to verify the system properly handles inaccurate distribution requests.

Test Case 1 Complete/Accurate distribution request

This test verifies the validation of a distribution request in which all required fields are complete and accurate.

Input:	User generation of a distribution request.
Output:	Status message and distribution request log.
Expected Results:	The status message states that the distribution request was received. The distribution request log contains an entry for the distribution request. The request is processed.

Test Case 2 Incomplete distribution request

This test verifies the system provides the user with an error message during generation of the distribution request if a required field is missing.

Input:	User generation of a distribution request. User attempts to leave a required field blank.
Output:	Status messages, error logs and responses to the user interface. User is unable to continue without inputting data into a required field.
Expected Results:	User is unable to submit a distribution request without completing all required fields.

Test Case 3 Complete/Inaccurate distribution request

This test verifies the system is able to respond to an invalid distribution request in which all required fields are complete but inaccurate.

Input:	User generation of a distribution request. User inputs invalid data into a required field.
Output:	Status messages, error logs and responses to the user interface. Response to user of inability to process request.
Expected Results:	Distribution request is submitted to the system with invalid data in a required field. System provides user a response indicating request can not be processed due to invalid data entry in a required field.

5.1.15.2.5 Sequence 5 - User Generation of Product Orders

The following tests verify the capability to generate complete, accurate product orders. Testing will also be performed to verify the system properly handles inaccurate product orders.

Test Case 1 Complete/Accurate Product Order

This test verifies the validation of a product order in which all required fields are complete and accurate.

Input:	User generation of a valid, complete product order for a known data product.
Output:	Status message and product order log.
Expected Results:	The status message states that the product order was received. The product order log contains an entry for the product order. The order is processed.

Test Case 2 Incomplete Product Order

This test verifies the system provides the user with an error message during generation of the product order if a required field is missing.

Input: User generation of a product order. User attempts to leave a required field blank.

Output: Status messages, error logs and responses to the user interface. User is unable to continue without inputting data into a required field.

Expected Results: User is unable to submit a product order without completing all required fields.

Test Case 3 Complete/Inaccurate Product Order

This test verifies the system is able to respond to an invalid product order in which all required fields are complete but inaccurate.

Input: User generation of a product order. User inputs invalid data into a required field.

Output: Status messages, error logs and responses to the user interface. Response to user of inability to process order.

Expected Results: Product order is submitted to the system with invalid data in a required field. System provides user a response indicating order can not be processed due to invalid data entry in a required field.

5.1.15.2.6 Sequence 6 - Receiving Request/Order Acknowledgments

The following series of tests verify the system's ability to provide the user with an acknowledgment to a distribution request or product order.

Test Case 1 Distribution Request Acknowledgment

This test verifies the capability to issue an acknowledgment for a distribution request.

Input: User generation of a distribution request for a known set of data.

Output: User receives acknowledgment for the distribution request. Event log contains entry for the generation of the distribution request and the receipt of the acknowledgment.

Expected Results: Distribution request is received by the system. An acknowledgment of the request is issued to the user.

Test Case 2 Product Order Acknowledgment

This test verifies the capability to issue an acknowledgment for a product order.

Input: User generation of a product order for a known product.

Output: User receives acknowledgment for the product order. Event log contains entry for the generation of the product order and the receipt of the acknowledgment.

Expected Results: Product order is received by the system. An acknowledgment of the request is issued to the user.

5.1.15.2.7 Sequence 7 - Utilizing the Hypertext Linked Help Capabilities

The following test case demonstrates the ability to access hypertext linked help data. Utilizing a workbench interface, the user is able to obtain help data on information regarding each DAAC, the data supported by that DAAC, and other data about the ECS system. Accessing help functions other than those previously tested (data dictionary, data server, LIM, and advertising service) are examined under this test case.

Test Case 1 Utilizing Hypertext Help

This test verifies the capability to obtain help information across a heterogeneous network. This test case is executed for each element (site) supported in Rel A.

Input: Workbench interface to a help function for each site available in Rel A.

Output: User able to obtain access to help function for each site under test. Information provided to the user is meaningful and will aid the user in navigating through the system. Event log contains an entry for the activation of the help function for the site under test.

Expected Results: User able to access site help function and navigate through the system under test.

5.1.15.2.8 Sequence 8 - Problem Reports

The following test case demonstrates the ability of the user to enter problem reports.

Test Case 1 Metadata Problem Reports

This test verifies the capability of the system to accept and process a metadata problem report from a user. The user utilizes the problem report form. Once the form is submitted, the tester verifies that the report is received by the responsible element.

Input: Problem report form.

Output: Report submitted.

Expected Results: Report is accepted and forwarded to the responsible element.

5.1.16 System Integration Build 4 (B13)

The System Integration Build 4 is an integrated build of the following system threads and builds:

System Integration Build 2
System Integration Build 3
CSMS System Monitoring Thread
User Services Thread

5.1.16.1 Build Objectives

The objectives of the system integration build 4 are:

- Provide user access to data and products to include data dictionary, LIM, advertising service, distribution requests and hypertext help functions.
- Provide access to the system monitoring tools.
- Provide access to the ECS desktop environment.
- Provide access to the search services.
- Provide access to the product generation functions.
- Provide network management tools.

5.1.16.2 Build Test Description

Testing is performed to verify the user access to data and products via the data dictionary, LIM, advertising service, and hypertext help functions. Tester will verify that the system monitoring tools respond to system faults and produce valid reports using network statistics. The ECS desktop environment is utilized to create searches, create and monitor product generation, and distribute products. Network management tools are tested.

Dependencies: (If Applicable)

In order to verify the objectives of the system integration build 4, the following interface and functional capabilities are required:

- Interface to data server, data dictionary, and LIM for data access.
- Interface to EOSView for data visualization.
- Interface to product generation system to create plans, manipulate production queues, and distribute products.
- Interface to network management tools.

Test Support Requirements

- Hardware:
  –
- Software:
  –
- Data:
  –

5.1.16.2.1 Sequence 1 - Data Access

The following tests verify that user privileges for data access to both products and raw data were properly defined during the user registration process. The tester creates user accounts for various types of users with varied data access privileges. The tester then tests the user's access to data in the system.

Test Case 1 User Registration

This test verifies that a user can register into the system and be granted different levels of data access based on the user registration. The ECS desktop is used to register users into the system and grant data access privileges. Hypertext help functions are used in filing out the registration forms.

Input: Log onto the ECS system and register users with different access privileges. Use hypertext help to display the information available in filling out user registration forms.

Output: The system generates messages to notify tester of the accounts created and data access privileges. Help messages aid in preparing registration forms. Event log and audit trails are updated with information about the new accounts.

Expected Results: The accounts are generated with varied data access privileges. Hypertext help information is displayed.

Test Case 2 Data Access Privileges

Using the ECS workbench and an account with limited privileges, the tester executes the product ordering service. From here, the tester requests access to data/products with restricted access. The request is denied or granted based on the accounts access privileges.

Input: Log onto the ECS system as users with varied access levels. Formulate a query, using the data/product ordering service, for known sets of restricted data/products. Submit the queries.

Output: User receives a notice that access to the data is denied/granted based on user access privileges. History log contains messages of the events that have transpired. The data access log is updated.

Expected Results: The user is granted access to data/product based on privileges defined in the user's profile. An audit record is created.

Test Case 3 Data Access Updates

This test case verifies that changes to data availability are reflected in the system inventory. Data purges, transfers from one site to another, unexpected loss of data and updates to data are verified. Data inventory reports are generated prior to updates to the inventory and after the updates are completed. The reports are verified for accuracy.

Input: Data inventory report, commands for data purges, data transfers, unexpected loss, and data updates.

Output: Data inventory reports.

Expected Results: The data inventory report reflects changes to the inventory.

5.1.16.2.2 Sequence 2 - Data Searching

The following series of tests verify the ability to use the ECS desktop and workbench services to locate and execute the data searching services. The tester, using the workbench, locates the search services icon and loads it on the workbench. From the workbench, the tester launches the search services application. From this application, the tester selects search criteria and submits the search. The returned results package is examined using the EOSView tool Fault testing is also verified.

Test Case 1 Install Search Services

This test verifies that the search services function can be found and installed on the ECS workbench. The tester uses the advertising service to locate the search service icon and move it to the workbench. This installation verifies that advertised services can be installed on the workbench. The icon is then double-clicked to bring u the search services window.

Input: ECS workbench, advertising service.

Output: Search services installed on the ECS workbench, search services window installed.

Expected Results: The search services icon is installed on the ECS Workbench. The search services window appears when the icon is executed.

Test Case 2 Search Services Execution

This test verifies that the user can utilize the search criteria to locate and retrieve data from the data server. The search criteria information is composed of information in the data server, LIM and data dictionary. Once the results package is returned, the EOSView tool is used to view the data.

Input: Various search criteria, EOSView tool.

Output: Returned search result packages, displayed data images.

Expected Results: The search criteria is entered and the returned results package is verified to ensure the correct information was returned. The EOSView tool is used to view the data which is returned.

Test Case 3 Search Services Faults

This test verifies that the system can recover from a network fault. The tester will have the network monitoring tool displayed on a workstation. A search scenario is started on another workstation. After the search request is submitted, the tester will induce a network fault which disables the communication link to the site containing the data. The network monitoring tool detects the fault, and generates alarm messages. Recovery procedures are begun. When the system regains the link to the site which was disabled, the tester verifies that the search request is completed as if no disruption had occurred. User abort of a search in progress is verified.

Input:	Various search criteria, tester induced network fault, user abort sequence.
Output:	Network fault detection, recover procedures correct communication problem, returned search result packages.
Expected Results:	The network fault is detected, recovery procedures are displayed and implemented, communication is reestablished, the returned results package is verified for completeness. User abort of a search request is successful.

5.1.16.2.3 Sequence 3 - Product Generation

This test case verifies the systems ability to create a processing plan, modify the production queue, and generate data products. Data is ingested into the system, a plan is created to utilize the data, the production queues are manipulated, and the products are produced. Fault testing is also verified.

Test Case 1 Process Plan Creation

This test case verifies that a valid request for creating a processing plan is acted upon by the system. The operator enters a create a plan message. The message must contain a start time, an end time, and planning criteria. The data for the plan is not in the system at the time of plan generation. The plan is then placed in the processing queue to wait for data availability. Planning criteria must contain a processing algorithm. The system verifies the parameters and generates the plan. Plan updates and cancellations are also verified. Reprocessing plan requests are verified.

Input:	Create a plan message with valid start and finish times, and valid planning criterion, valid update and cancellation commands.
Output:	The system validates the plan and sends messages to operational segments.
Expected Results:	The plan is placed in the processing queue to await data availability. Updates and cancellation requests are implemented.

Test Case 2 TRMM Ancillary Preprocessing and Ingest

This test case verifies that when data is received into the system that the system recognizes the data type and performs preprocessing prior to ingesting the data into the system. This test case uses each type of TRMM ancillary data which requires preprocessing. The tester manually alters the routing of data sets to physical storage locations. A data availability notice is sent to the production planning system upon completion of the ingest. One of the data ingested is the one that test case 1 is waiting for. Data transfer cancellation and delay notifications are verified.

Input:	Each type of TRMM ancillary data which requires preprocessing, data transfer cancellation and delay notifications.

Output:	Data availability notice, archive inventory reflects addition of new data to the data server. Local inventory logs contains entries for each data ingest.
Expected Results:	Data type is determined by the ingest service, preprocessing is performed, and the data is ingested into the system. A data availability notice is sent to production planning process for each data ingest. A data availability notice is sent to the production planning system, for each data ingest, to advertise that the data is available for product generation. Data transfer cancellation and delay notifications are acted upon by the system. Whenever the data is not going to arrive due to data transfer cancellation or delay notification, the appropriate alarm messages are generated. The inventory is updated upon successful ingest.

Test Case 3 Queue Fault Detection

This test case verifies that the production queue can be manipulated and fault detection/recovery is operational. The plan created in test case 1 is now able to be executed. The tester populates the queue with additional production plans. Once the queue is populated, the tester changes the status of the plan generated in test case one to execute. Once the plan begins execution, the tester induces a system fault which prohibits the executing job from completing. The system monitoring tool detects the fault and generates alarms. The job is stopped automatically and placed back in the production queue to wait for the problem to be corrected. The job is restarted once recovery procedures are completed. The job completes and the product is advertised.

Input:	Production plan from test case 1, production queue manipulation tool, tester induced fault, network monitoring tool, HP OpenView, recovery procedures.
Output:	Fault detection alarms, generated product.
Expected Results:	The production queue allows user manipulation. The HP OpenView monitoring tool detects the fault and records it. The plan is rescheduled after recovery and the plan executes as expected. The product is advertised upon completion.

Test Case 4 Resource Fault Detection

This test case verifies that the production queue can tolerate resource failure. As one of the plans created in test case 2 is executing, the tester induces a resource failure on one of the resources needed for plan execution. The system monitoring tool detects the fault and generates alarms. The job is stopped automatically and placed back in the production queue to wait for the resource to become available. The job is then restarted once the resource is available. The job completes and the product is advertised.

Input:	Production plan from test case 1, production queue manipulation tool, tester induced fault, network monitoring tool, HP OpenView, recovery procedures.

Output: Fault detection alarms, generated product.

Expected Results: The production queue allows user manipulation. The HP OpenView monitoring tool detects the resource fault and records it. The plan is rescheduled after recovery and the plan executes as expected. The product is advertised upon completion.

Test Case 5 Distribution request

This test verifies the validation of a distribution request in which all required fields are complete and accurate. The distribution request is for the advertised product from test case 3.

Input: User generation of a distribution request.

Output: Status message and distribution request log.

Expected Results: The status message states that the distribution request was received. The distribution request log contains an entry for the distribution request. The request is processed.

5.1.16.2.4 Sequence 4 - Product Orders

The following tests verify the capability to generate complete, accurate product orders. Testing will also be performed to verify the system properly handles inaccurate product orders.

Test Case 1 Install Product Ordering Services

This test verifies that the product ordering service can be found and installed on the ECS workbench. The tester uses the advertising service to locate the product ordering service icon and move it to the workbench. This installation verifies that advertised services can be installed on the workbench. The icon is then double-clicked to bring up the search services window.

Input: ECS workbench, advertising service.

Output: Product ordering service installed on the ECS workbench, product ordering service window comes up.

Expected Results: The product ordering service icon is installed on the ECS Workbench. The product ordering service window appears when the icon is executed.

Test Case 2 Product Order

This test verifies the validation of a product order in which all required fields are complete and accurate. Both standing and retrospective product orders are verified. Once the orders are the system, changes to standing orders are verified.

Input: User generation of a valid, complete product order for a known data product, changes to standing orders.

Output: Status message and product order log.

Expected Results: The status message states that the product order was received. The product order log contains an entry for the product order. The order is processed. Changes to standing orders are verified and implemented.

Test Case 3 Product Order Fault Detection

This test case verifies that fault detection/recovery is operational in conjunction with product ordering. Once the product order in test 2 has been accepted by the system, the tester induces an error an the site in which the product resides. The system monitoring tool detects the fault and generates alarms. Recovery tools are executed and the product order is processed.

Input: Product order from test case 2, tester induced fault, network monitoring tool, HP OpenView, recovery procedures.

Output: Fault detection alarms, generated product.

Expected Results: The network fault is detected and the recovery procedures are executed. The product order is processed as though no error has occurred.

5.1.16.2.5 Sequence 5 - Network Management

This test case verifies the network management functionality. The system' ability to monitor and control the environment, perform network reconfiguration, and generate reports to use for study of network performance are verified.

Test Case 1 Monitor and Control

This test case verifies that the system can be monitored and controlled. The HP OpenView tool is used to monitor communication software errors, configuration errors, communications hardware errors, protocol errors, performance degradation conditions, and telecommunications errors and failures. The tools ability to set, view, and change alert thresholds values, enable and disable alert notifications, event reports, and event logs is verified. Debugging and packet tracing utilities are demonstrated. Alternative communications protocols are tested.

Input: Tester induced error conditions, commands to set, change, and view alert threshold values, alert notification, and event reports and logs.

Output: The monitoring tool reflects system faults induced into the system. Displays and reports reflect changes.

Expected Results: All tools, reports, and utilities respond to tester induced stimuli.

Test Case 2 Network Reconfiguration

This test case verifies that the system network can be re configured without disturbance to on going system performance. The tester monitors system functions while re configuring the network. The reconfiguration is transparent to users at the time.

Input:	Network reconfiguration commands.
Output:	Network re configured, monitoring display updates.
Expected Results:	The network is re configured without user interruption.

Test Case 3 Report Generation

This test case verifies that all system reports can be generated, interactively or on a scheduled basis, to provide information concerning network management, performance management, configuration management, and fault management. The report output can be directed to consoles, disk files, and/or printers. Network trend analysis reports are also generated.

Input:	Report generation commands.
Output:	Reports directed to consoles, disk files, and/or printers.
Expected Results:	The network reports are produced and contain verifiable information.

5.1.17 TRMM Ingest Thread (T15-01)

5.1.17.1 Thread Objectives

The objectives of the TRMM ingest thread are:

- TRMM data ingest and validation.
- TRMM metadata validation and extraction.
- TRMM data staging.

5.1.17.2 Thread Description

The TRMM ingest thread verifies that TRMM data can be ingested into the ECS.

Testing is performed to verify the capability to ingest all types of TRMM data. Data validation is verified, metadata validation and/or generation is performed, and data reformatting and/or data compression is verified. Data ingest is monitored using a monitoring tool. Generation of an ingest completion notification is verified.

Dependencies: (If Applicable)

In order to verify the objectives of the TRMM ingest thread, the following interface and functional capabilities are required:

- Interface to the data server/archive to perform data storage.
- Interface to receive TRMM platform lower level data and higher level data products.

Test Support Requirements

- Hardware:
  - operational system

- Software:
  - operational software
  - simulated user interface for data ingest
- Data:
  - VIRS, CERES, LIS, PR, TMI data

5.1.17.2.1 Sequence 1 - VIRS Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest VIRS data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 VIRS L0-L1 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of VIRS data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: VIRS L0 and L1 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: VIRS data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 VIRS L0-L1 Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of VIRS data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid VIRS L0 and L1 data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: VIRS data ingest terminated and originator receives retransmission notification.

Test Case 3 VIRS Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of VIRS product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the VIRS data products supported in Rel A (N0074, N0037c, G-71).

Input: VIRS data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: VIRS product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 VIRS Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with VIRS data and to provide a status of the transaction.

Input: VIRS data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid VIRS Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting VIRS data.

Input: VIRS data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 VIRS Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated VIRS product. Originator is notified of the re-generation of metadata for the associated product data.

Input: VIRS data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 VIRS Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of VIRS data.

Input: VIRS data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.17.2.2 Sequence 2 - CERES Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest CERES data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 CERES L0-L1A Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of CERES data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: CERES L0 and L1A data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: CERES data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 CERES L0-L1A Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of CERES data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid CERES L0 and L1A data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: CERES data ingest terminated and originator receives retransmission notification.

Test Case 3 CERES Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of CERES product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the CERES data products supported in Rel A (CASTR, N0039h, CCRHA, CCIDA, CER00, CER01, etc).

Input: CERES data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: CERES product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 CERES Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with CERES data and to provide a status of the transaction.

Input: CERES data resident in the staging area, status message, and an ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid CERES Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting CERES data.

Input: CERES data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 CERES Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated CERES product. Originator is notified of the re-generation of metadata for the associated product data.

Input: CERES data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 CERES Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of CERES data.

Input: CERES data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.17.2.3 Sequence 3 - LIS Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest LIS data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 LIS L0 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of LIS data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: LIS L0 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: LIS data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 LIS L0 Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of LIS data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid LIS L0 data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: LIS data ingest terminated and originator receives retransmission notification.

Test Case 3 LIS Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of LIS product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the LIS data products supported in Rel A (LIS01, LIS02, LIS03, LIS04, LIS05, LIS06, LIS07, LIS08, LIS09, etc).

Input: LIS data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: LIS product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 LIS Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with LIS data and to provide a status of the transaction.

Input: LIS data resident in the staging area, status message, and an ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid LIS Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting LIS data.

Input: LIS data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 LIS Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated LIS product. Originator is notified of the re-generation of metadata for the associated product data.

Input: LIS data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-

processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 LIS Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of LIS data.

Input: LIS data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.17.2.4 Sequence 4 - PR Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest PR data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 PR L0 - L1 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of PR data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: PR L0 and L1 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: PR data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 PR L0 Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of PR data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid PR L0 and L1 data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: PR data ingest terminated and originator receives retransmission notification.

Test Case 3 PR Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of PR product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the PR data products supported in Rel A (M-65, M66, M67, M-68, M-69, M-70, G-71, G-72 etc).

Input: PR data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: PR product data and metadata ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 PR Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with PR data and to provide a status of the transaction.

Input: PR data resident in the staging area, status message, and an ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid PR Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting PR data.

Input: PR data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 PR Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated PR product. Originator is notified of the re-generation of metadata for the associated product data.

Input: PR data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 PR Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of PR data.

Input: PR data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.17.2.5 Sequence 5 - TMI Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest TMI data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 TMI L0 - L1 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of TMI data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: TMI L0 and L1 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: TMI data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 TMI L0 Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of TMI data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid TMI L0 and L1 data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: TMI data ingest terminated and originator receives retransmission notification.

Test Case 3 TMI Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of TMI product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the TMI data products supported in Rel A (M-71, M-72, M-73, M-74, M-75, G-71, G-72, etc).

Input: TMI data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: TMI product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 TMI Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with TMI data and to provide a status of the transaction.

Input: TMI data resident in the staging area, status message, and an ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid TMI Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting TMI data.

Input: TMI data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 TMI Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0; date and time of storage; physical location; data check status; unique format identifiers) for an associated TMI product. Originator is notified of the re-generation of metadata for the associated product data.

Input: TMI data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 TMI Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of TMI data.

Input: TMI data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.17.2.6 Sequence 6 - Archiving Staged Data

The following series of tests verify the ability to archive data in the temporary staging area to a permanent archive. Testing will verify the ability to recover from an error condition during the permanent archive execution.

Test Case 1 Archiving Data in the Staging Area

This test verifies the interface between ingest and the data server to initiate permanent archiving of data in the staging area.

Input: Data resident in the staging area. Initiate permanent archiving.

Output: Ingest log contains an entry for the call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Data in the staging area written to permanent archive, disk space allocated for temporary staging relinquished.

Test Case 2 Error Condition Attempting to Archive Data

This test verifies the capability to respond to an error condition encountered while performing the write to the permanent archive. Testing will verify that no data is lost as a result of the error condition.

Input: Data resident in the staging area, initiate permanent archiving, produce an error condition during the archive (i.e., power off data staging disk).

Output: Ingest log contains an entry for the call to the data server to initiate data storage, error condition encountered, and failed archive attempt. Data remains in the staging area. Incomplete data set not saved to the archive.

Expected Results: Data remains in the staging area, disk space not de-allocated. Data set not contained in the archive inventory.

5.1.17.2.7 Sequence 7 - Ingesting Ancillary/Calibration Data Sets

The following series of tests verify the ability to ingest and archive ancillary and calibration data sets.

Test Case 1 Ingesting and Archiving Ancillary Data Set

This test verifies the ability to ingest ancillary data sets and perform permanent archiving.

Input: Ancillary data set to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of ancillary data set ingest, call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Ancillary data ingested, written to the staging area, written to permanent archive, and the staging disk space de-allocated.

Test Case 2 Ingesting and Archiving Calibration Data Set

This test verifies the ability to ingest calibration data sets and perform permanent archiving.

Input: Calibration data set to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of calibration data set ingest, call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Calibration data ingested, written to the staging area, written to permanent archive, and the staging disk space de-allocated.

5.1.17.2.8 Sequence 8 - Ingesting Algorithms

The following series of tests verify the ability to ingest algorithms, validate header content/descriptors, write the data to permanent archive, and generate an ingest completion notice to the originator of the ingest.

Test Case 1 Algorithm Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of the algorithm, initiate data ingestation, and verify header content/descriptor.

Input: Algorithm to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of algorithm ingest and the validation of the header content/descriptor information. Data contained in the temporary staging area.

Expected Results: Algorithm data ingested, header content/descriptor information validated, and written to the staging area.

5.1.17.2.9 Sequence 9 - Ingesting Non-EOS Correlative Data

The following series of tests verify the ability to indicate its readiness to accept data, ingest Non-EOS correlative data, validate data integrity, perform pre-processing if necessary, write the data to permanent archive, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 Non-EOS Correlative Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of Non-EOS correlative data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: Non-EOS correlative data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: Non-EOS correlative data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 Non-EOS Correlative Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of Non-EOS correlative data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid Non-EOS correlative data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: Non-EOS correlative data ingest terminated and originator receives retransmission notification.

Test Case 3 Non-EOS Correlative Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of Non-EOS product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the Non-EOS Correlative data products supported in Rel A.

Input: Non-EOS Correlative data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: Non-EOS Correlative product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 4 Non-EOS Correlative Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received with Non-EOS correlative data and to provide a status of the transaction.

Input: Non-EOS correlative data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, completion of metadata validation, and call to the data server to initiate data storage.

Expected Results: Metadata is validated, status message received, call to the data server to initiate permanent data archiving is performed.

Test Case 5 Invalid Non-EOS Correlative Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting Non-EOS Correlative data.

Input: Non-EOS Correlative data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

Test Case 6 Non-EOS Correlative Metadata - Reproduce

This test verifies the ability to reproduce metadata (at a minimum: unique granule id for L0, date and time of storage; physical location; data check status; unique format identifiers) for an associated Non-EOS Correlative product. Originator is notified of the re-generation of metadata for the associated product data.

Input: Non-EOS correlative data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields). Ability to perform pre-processing to reproduce product metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, initiation of pre-processing to reproduce the metadata, completion of metadata generation, notification to the originator of metadata re-generation, and call to the data server to write the data to permanent archive.

Expected Results: Metadata validation fails, metadata re-created, data written to permanent archive, and staging disk space de-allocated.

Test Case 7 Non-EOS Correlative Ingest Pre-processing

This test verifies the ability to perform any required pre-processing of Non-EOS Correlative data.

Input: Non-EOS Correlative data resident in the staging area, ability to perform pre-processing.

Output: Ingest log contains an entry for the initiation of pre-processing, completion of pre-processing, and call to the data server to initiate data storage.

Expected Results: Pre-processing is initiated and completed and call to the data server to initiate permanent data archiving is performed.

5.1.18 AM-1 Ingest Thread (T15-02)

5.1.18.1 Thread Objectives

The objectives of the AM-1 ingest thread are as follows:

- AM-1 data ingest and validation.
- AM-1 metadata validation and extraction.
- AM-1 data staging.

5.1.1.18.2 Thread Description

Testing is performed to verify early interface testing requirements for ingesting of AM-1 data and metadata. Testing verifies the requirement to ingest and stage all AM-1 instrument data types (ASTER, MODIS, MISR, MOPITT).

Testing is also performed to verify early interface requirements for ingesting Landsat 7 data.

Dependencies: (If Applicable)

In order to verify the objectives of the AM-1 ingest thread, the following interface and functional capabilities are required:

- Interface to receive simulated AM-1 platform data.
- Interface to receive simulated Landsat 7 platform data.

Test Support Requirements

- Hardware:
  - Workstation for ECS Management/Monitor Utility
- Software:

– Simulated interfaces (as needed)

- Data:
  - Simulated AM-1 L0-L4 data and metadata
  - Simulated AM-1 L0-L4 data products and metadata
  - Simulated Landsat 7 product data
  - Ancillary data

5.1.18.2.1 Sequence 1 - ASTER Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest ASTER data, validate data integrity, write the data to the staging area, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 ASTER L0 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of ASTER L0 data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: ASTER L0 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: ASTER L0 data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 ASTER L1A Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of ASTER L1A data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: ASTER L1A data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: ASTER L1A data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 3 ASTER L0-L1A Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of ASTER data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid ASTER L0 and L1A data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: ASTER data ingest terminated and originator receives retransmission notification.

Test Case 4 ASTER Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of ASTER product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the ASTER data products to be supported (AST05, AST06, AST07, AST08, N0009, etc).

Input: ASTER data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: ASTER product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 5 ASTER Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received (at a minimum: presence of required fields; error-free input; correctness of the data set granule size; and other checks as required) with ASTER data and to provide a status of the transaction.

Input: ASTER data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, and completion of metadata validation.

Expected Results: Metadata is validated and status message is received.

Test Case 6 Invalid ASTER Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting ASTER data.

Input: ASTER data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

5.1.18.2.2 Sequence 2 - MODIS Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest MODIS data, validate data integrity, write the data to the staging area, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 MODIS L0 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MODIS L0 data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MODIS L0 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MODIS L0 data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 MODIS L1A Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MODIS L1A data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MODIS L1A data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MODIS L1A data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 3 MODIS L1B Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MODIS L1B data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MODIS L1B data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MODIS L1B data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 4 MODIS L0-L1A-L1B Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of MODIS data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid MODIS L0, L1A, and L1B data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: MODIS data ingest terminated and originator receives retransmission notification.

Test Case 5 MODIS Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MODIS product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the MODIS data products to be supported (MOD05, MOD21, MOD31, MOD32, MOD34, MOD38, etc).

Input: MODIS data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: MODIS product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 6 MODIS Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received (at a minimum: presence of required fields; error-free input; correctness of the data set granule size; and other checks as required) with MODIS data and to provide a status of the transaction.

Input: MODIS data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, and completion of metadata validation.

Expected Results: Metadata is validated and status message is received.

Test Case 7 Invalid MODIS Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting MODIS data.

Input: MODIS data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

5.1.18.2.3 Sequence 3 - MISR Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest MISR data, validate data integrity, write the data to the staging area, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 MISR L0 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MISR L0 data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MISR L0 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MISR L0 data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 MISR L1A Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MISR L1A data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MISR L1A data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MISR L1A data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 3 MISR L1B Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MISR L1B data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MISR L1B data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MISR L1B data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 4 MISR L0-L1A-L1B Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of MISR data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input: Invalid MISR L0, L1A, and L1B data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.

Expected Results: MISR data ingest terminated and originator receives retransmission notification.

Test Case 5 MISR Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MISR product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the MISR data products to be supported (MIS04, MIS03, MIS04, MIS07, etc).

Input: MISR data product, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.

Expected Results: MISR product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 6 MISR Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received (at a minimum: presence of required fields; error-free input; correctness of the data set granule size; and other checks as required) with MISR data and to provide a status of the transaction.

Input: MISR data resident in the staging area, status message, and the ability to verify metadata.

Output: Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, and completion of metadata validation.

Expected Results: Metadata is validated and status message is received.

Test Case 7 Invalid MISR Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting MISR data.

Input: MISR data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

5.1.18.2.4 Sequence 4 - MOPITT Ingest

The following series of tests verify the ability to indicate its readiness to accept data, ingest MOPITT data, validate data integrity, write the data to the staging area, generate an ingest completion notice to the originator of the ingest, and prepare data availability notices for required interfaces.

Test Case 1 MOPITT L0 Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MOPITT L0 data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MOPITT L0 data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MOPITT L0 data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 2 MOPITT L1B Data Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MOPITT L1B data, initiate data ingestation, verify data integrity, notify originator of ingest completion, and prepare data availability notices for required interfaces.

Input: MOPITT L1B data, ingest monitoring tool, ability to perform data validation.

Output: Ingest log contains an entry for the initiation of data ingest, the validation of the data, the notification of ingest completion, and the generation of data availability notices for required interfaces. Data contained in the temporary staging area.

Expected Results: MOPITT L1B data ingested, written to the staging area, originator of ingest notified of completion, and data availability notices are generated for the required interfaces.

Test Case 3 MOPITT L0-L1B Data Ingest Initiation - Invalid

This test verifies the ability to acknowledge the receipt of MOPITT data, verify that the data received came from an approved/authorized source, initiate data ingestation, verify

data integrity, terminate data ingestation upon failure of data integrity check, and issue retransmission request to originator.

Input:	Invalid MOPITT L0 and L1B data from an approved/authorized source (i.e., required header fields contain 'null' values), ingest monitoring tool, ability to perform data validation.
Output:	Ingest log contains an entry for the initiation of data ingest, initiation of data validation, failure of data integrity check, termination of data ingest, and generation of retransmission request. No data exists in the temporary staging area.
Expected Results:	MOPITT data ingest terminated and originator receives retransmission notification.

Test Case 4 MOPITT Product Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of MOPITT product data and metadata, initiate data ingestation, verify data integrity, and notify originator of ingest completion. This test case is executed for each of the MOPITT data products to be supported (MOP00, MOP02, MOP03, MOP07, etc.).

Input:	MOPITT data product, ingest monitoring tool, ability to perform data validation.
Output:	Ingest log contains an entry for the initiation of data ingest, the validation of the data integrity, and notification of ingest completion. Data contained in the temporary staging area.
Expected Results:	MOPITT product data and metadata are ingested, written to the staging area, and originator of ingest notified of completion.

Test Case 5 MOPITT Metadata Validation and Transaction Status

This test verifies the ability to validate the corresponding metadata received (at a minimum: presence of required fields; error-free input; correctness of the data set granule size; and other checks as required) with MOPITT data and to provide a status of the transaction.

Input:	MOPITT data resident in the staging area, status message, and the ability to verify metadata.
Output:	Ingest log contains an entry for the initiation of metadata validation, status message generated and logged, and completion of metadata validation.
Expected Results:	Metadata is validated and status message is received.

Test Case 6 Invalid MOPITT Metadata - Notify Originator

This test verifies the ability to notify the originator of invalid metadata received for the supporting MOPITT data.

Input: MOPITT data resident in the staging area, ability to verify metadata. Metadata contains invalid data (i.e., missing required fields).

Output: Ingest log contains an entry for the initiation of metadata validation, failure of metadata validation, and generation of retransmission request to the originator.

Expected Results: Metadata validation fails, originator notified.

5.1.18.2.5 Sequence 5 - Landsat 7 Early Interface Testing

The following series of tests verify the ability to support early interface testing for Landsat 7.

Test Case 1 Staging L7 Product Data

This test verifies the ability to acknowledge the receipt of Landsat 7 product data. This test case is executed for each of the L7 data product to be supported (L0R, E-1, E-7, E-8, E-9, E-10, etc.).

Input: L7 product data, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of data ingest and the successful staging of the L7 data product.

Expected Results: L7 data ingested and written to the staging area.

5.1.18.2.6 Sequence 6 - Ingesting Ancillary/Calibration Data Sets

The following series of tests verify the ability to ingest and stage ancillary and calibration data sets.

Test Case 1 Data Staging of Ancillary Data Set(s)

This test verifies the ability to ingest ancillary data sets and perform data staging.

Input: Ancillary data set to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of ancillary data set ingest, call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Ancillary data ingested and written to the staging area.

Test Case 2 Data Staging of Calibration Data Set(s)

This test verifies the ability to ingest calibration data sets and perform staging.

Input: Calibration data set to ingest, ingest monitoring tool.

Output: Ingest log contains an entry for the initiation of calibration data set ingest, call to the data server to initiate data storage, completion of data storage, and generation of completion notice.

Expected Results: Calibration data ingested and written to the staging area.

5.1.18.2.7 Sequence 7 - Ingesting Algorithms

The following series of tests verify the ability to ingest algorithms, validate header content/descriptors, write the data to staging disk, and generate an ingest completion notice to the originator of the ingest.

Test Case 1 — Algorithm Ingest Initiation - Validation

This test verifies the ability to acknowledge the receipt of the algorithm, initiate data ingestation, and verify header content/descriptor.

Input:	Algorithm to ingest, ingest monitoring tool.
Output:	Ingest log contains an entry for the initiation of algorithm ingest and the validation of the header content/descriptor information. Data contained in the temporary staging area.
Expected Results:	Algorithm data ingested, header content/descriptor information validated, and written to the staging area.

5.1.19 Two-Way V0 Interoperability Thread (T15-03)

5.1.19.1 Thread Objectives

The objectives of the two-way V0 Interoperability Thread are as follows:

Provide both outgoing and incoming interoperability.

Provide bi-directional on-line interfacing between ECS and V0.

5.1.19.2 Thread Test Description

To demonstrate two-way interoperability, ECS and V0 must exchange metadata listings and data access services in order for each system to advertise data holdings to the other system. Updates to any metadata holdings must be made available to all clients to fully demonstrate two-way interoperability. Log file messages will be generated indicating successful addition of metadata and successful advertisement to all clients.

Also, ECS must allow users (testers) to browse, search, and order data products from V0 and vice versa, in order to further demonstrate two-way interoperability. The V0 network and NSI will be monitored to verify all authorization and authentication between systems is transparent to the user (tester).

Dependencies: (If Applicable)

In order to verify the objectives of the two-way V0 interoperability thread, the following interfaces and functional capabilities are required:

- Each system must contain inventory, guide, and browse metadata which can be manipulated to verify advertisements across systems.
- Access to functionality to manipulate the inventory, guide, and browse metadata and add/delete new metadata from each system.

- Interface with V0/NSI for access to data holdings from ECS and V0 for search requests.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  - COTS /public domain software
  - Version 0 IMS
- Data:
  - TBS

5.1.19.2.1 Sequence 1 - Inventory Metadata Update

This sequence of tests will verify that when the inventory metadata is updated (in either V0 controlled inventory or ECS controlled inventory) that all clients, in both V0 and ECS, are notified of the changes. This notification will demonstrate the interoperability between V0 and ECS.

Test Case 1 ECS altered Metadata

The tester will construct a search request for ECS inventory metadata from a V0 client. The tester will submit the request, and verify the search package results. The tester will then alter the metadata, and submit the altered metadata to ECS from a V0 client. The tester will re-submit the initial search request, which should result in a null match (metadata has changed from original state). The tester will then construct a search request, specific to the modifications made to the original inventory metadata, and submit the request. The final search package results will verify the inventory metadata has been updated.

Input: Search request, valid inventory metadata update transaction.

Output: Search package results (initial and follow-up after metadata update), message log indicating inventory metadata schema change and successful advertising to all clients of change.

Expected Results: The final search package should reflect the modified inventory metadata schema.

Test Case 2 V0 altered Metadata

The tester will construct a search request for V0 inventory metadata from a ECS client. The tester will submit the request, and verify the search package results. The tester will then alter the metadata, and submit the altered metadata to V0 from a ECS client. The tester will re-submit the initial search request, which should result in a null match (metadata has changed from original state). The tester will then construct a search request, specific to the modifications made to the original inventory metadata, and submit

the request. The final search package results will verify the inventory metadata has been updated.

Input: Search request, valid inventory metadata update transaction.

Output: Search package results (initial and follow-up after metadata update), message log indicating inventory metadata schema change and successful advertising to all clients of change.

Expected Results: The final search package should reflect the modified inventory metadata schema.

5.1.19.2.2 Sequence 2 - Guide Information Update

This sequence of tests will verify that when the guide information inventory is updated (in either V0 controlled inventory or ECS controlled inventory) that all clients, in both V0 and ECS, are notified of the changes. This notification will demonstrate the interoperability between V0 and ECS.

Test Case 1 ECS altered Guide Information

The tester will construct a search requesting ECS guide information from a V0 client. The tester will submit the request, and verify the search package results. The tester will then alter the guide information, and submit the altered information to ECS from a V0 client. The tester will re-submit the initial search request, which should result in a null match (guide information has changed from original state). The tester will then construct a search request, specific to the modifications made to the original guide information, and submit the request. The final search package results will verify the inventory metadata has been updated.

Input: Search request, valid guide information update transaction.

Output: Search package results (initial and follow-up after guide information update), message log indicating guide information schema change and successful advertising to all clients of change.

Expected Results: The final search package result should reflect the modified guide information schema.

Test Case 2 V0 altered Guide Information

The tester will construct a search requesting V0 guide information from a ECS client. The tester will submit the request, and verify the search package results. The tester will then alter the guide information, and submit the altered information to V0 from a ECS client. The tester will re-submit the initial search request, which should result in a null match (guide information has changed from original state). The tester will then construct a search request, specific to the modifications made to the original guide information, and submit the request. The final search package results will verify the inventory metadata has been updated.

Input: Search request, valid guide information update transaction.

Output: Search package results (initial and follow-up after guide information update), message log indicating guide information schema change and successful advertising to all clients of change.

Expected Results: The final search package result should reflect the modified guide information schema.

5.1.19.2.3 Sequence 3 - Browse Product Inventory Update

This sequence of tests will verify that when the browse product inventory is updated (in either V0 controlled inventory or ECS controlled inventory) that all clients, in both V0 and ECS, are notified of the changes. This notification will demonstrate the interoperability between V0 and ECS.

Test Case 1 ECS altered Browse Products Inventory Information

The tester will construct a search requesting ECS browse product information from a V0 client. The tester will submit the request, and verify the search package results. The tester will then alter the browse product information, and submit the altered information to ECS from a V0 client. The tester will re-submit the initial search request, which should result in a null match (information has changed from original state). The tester will then construct a search request, specific to the modifications made to the original browse product information, and submit the request. The final search package results will verify the browse product inventory has been updated.

Input: Search request, valid browse product information update transaction.

Output: Search package results (initial and follow-up after browse product information update), message log indicating browse product information schema change and successful advertising to all clients of change.

Expected Results: The final search package result should reflect the modified browse product information schema.

Test Case 2 V0 altered Browse Products Inventory Information

The tester will construct a search requesting V0 browse product information from a ECS client. The tester will submit the request, and verify the search package results. The tester will then alter the browse product information, and submit the altered information to V0 from a ECS client. The tester will re-submit the initial search request, which should result in a null match (information has changed from original state). The tester will then construct a search request, specific to the modifications made to the original browse product information, and submit the request. The final search package results will verify the browse product inventory has been updated.

Input: Search request, valid browse product information update transaction.

Output: Search package results (initial and follow-up after browse product information update), message log indicating browse product

information schema change and successful advertising to all clients of change.

Expected Results: The final search package result should reflect the modified browse product information schema.

5.1.19.2.4 Sequence 4 - New Data Availability

This sequence of tests will verify that when new data is added to a data site, that the inventory is updated to reflect the new data. The test will verify that both V0 and ECS are informed of updates to the existing data schemas.

Test Case 1 Addition of ECS Data

The tester will construct a search requesting ECS data from a V0 client. The tester will submit the request, and verify the search package results. The tester will then proceed to ingest data into the ECS system. After the data is ingested, the tester will submit the same search request, from the V0 client. The returned result package should reflect the newly added ECS data.

Input: Search request, data to ingest.

Output: Search package results (initial and follow-up after ingesting data), message log indicating data ingested and successful advertising to all clients of new data.

Expected Results: The final search package result should reflect the newly ingested data.

Test Case 2 Addition of V0 Data

The tester will construct a simple search requesting V0 data (same type that will be added) from an ECS client. The tester will submit the request, and verify the search package results. The tester will then proceed to ingest data into the V0 system. After the data is ingested, the tester will submit the same search request, from the ECS client. The returned result package should reflect the newly added V0 data.

Input: Search request, data to ingest.

Output: Search package results (initial and follow-up after ingesting data), message log indicating data ingested and successful advertising to all clients of new data.

Expected Results: The final search package result should reflect the newly ingested data.

5.1.19.2.5 Sequence 5 - Removal of Data

This sequence of tests will verify that when data is removed from a data site, that the inventory is updated to reflect the loss of data. The test will verify that both V0 and ECS are informed of updates to the existing data schemas.

Test Case 1 Removal of ECS Data

The tester will construct a search requesting ECS data from a V0 client. The tester will submit the request, and verify the search package results. The tester will then proceed to remove data from the ECS system. After the data is deleted, the tester will submit the same search request, from the V0 client. The search result should reflect the fact that the ECS data was deleted.

Input: Search request, data to delete.

Output: Search package results (initial and follow-up after deleting data), message log indicating data deleted.

Expected Results: The final search package result should reflect the deleted data.

Test Case 2 Removal of V0 Data

The tester will construct a search requesting V0 data from an ECS client. The tester will submit the request, and verify the search package results. The tester will then proceed to delete data from the V0 system. After the data is deleted, the tester will submit the same search request, from the ECS client. The search result should reflect the fact that the V0 data was deleted.

Input: Search request, data to ingest.

Output: Search package results (initial and follow-up after ingesting data), message log indicating data ingested and successful advertising to all clients of new data.

Expected Results: The final search package result should reflect the newly ingested data.

5.1.19.2.6 Sequence 6 - Schema Updates

This sequence of tests verifies that when data schemas are updated, the information is advertised correctly to reflect the update. The V0 guide schema and the ECS browse product schema are updated in these tests. Interoperability is successful when V0 information is updated and advertised to ECS as well as ECS data being updated and advertised to the V0 system.

Test Case 1 Two-Way Interoperabilty - Land Guide Data

The tester will construct a request for land guide data from the ECS client to be submitted to the V0 data depository. The tester will use the core metadata search techniques to build the search request. Once the request is submitted the tester will validate the returned search package result. The tester will then update the guide schema information in the V0 data depository. Once the information is updated, the tester will resubmit the same search request for land guide data. The returned search package result should contain the updated information.

Input: Search request, V0 guide schema update.

Output: Search package results, message logs noting the updates to the V0 guide schema as well as the advertising of the changes.

Expected Results: The initial result package should contain all the V0 information which corresponds to the guide metadata search request. The updating of the V0 guide schema should be broadcast with messages. The final result package should contain the updated information.

Test Case 2 Two-Way Interoperabilty - Ocean Browse Product Data

The tester will construct a request for ocean browse product data from the V0 client to be submitted to the ECS data depository. The tester will use the core metadata search techniques to build the search request. Once the request is submitted the tester will validate the returned search package result. The tester will then update the browse product schema information in the ECS data depository. Once the information is updated, the tester will resubmit the same search request for land ocean browse product data. The returned search package result should contain the updated information.

Input: Search request, ECS browse product schema update.

Output: Search package results, message logs noting the updates to the ECS browse product data schema as well as the advertising of the changes.

Expected Results: The initial result package should contain all the ECS information which corresponds to the ocean browse product metadata search request. The updating of the ECS browse product schema should be broadcast with messages. The final result package should contain the updated information.

5.1.19.2.7 Sequence 7 - ECS/V0 User Access

This sequence of tests will verify the interface between ECS and V0 is transparent to the tester. The tester will request metadata specific to ECS from V0, and from ECS to V0 to verify the user authentication process is transparent.

Test Case 1 ECS User Requesting V0 Metadata

The tester will construct a search specific to the V0 inventory metadata from an ECS client. The tester will submit the request to ECS and select the option to continue the search at V0.

Input: Search request.

Output: Option to limit search within ECS system, or continue search to V0.

Expected Results: Search request is forwarded to V0 for further processing. User authentication from ECS to V0 should be transparent to the tester.

Test Case 2 V0 User Requesting ECS Metadata

The tester will construct a simple search specific to the ECS inventory metadata from a V0 client. The tester will submit the request to V0 and select the option to continue the search at ECS.

Input: Search request.

Output: Option to limit search within V0 system, or continue search to ECS.

Expected Results: Search request is forwarded to ECS for further processing. User authentication from V0 to ECS should be transparent to the tester.

5.1.19.2.8 Sequence 8 - ECS (User)/V0 Information Management

These series of tests will verify the tester is able to interface with V0 from ECS to perform inventory and guide searches, browse requests, and order data products.

Test Case 1 V0 Inventory Search

This test verifies the tester is able to access and search V0 data products. The tester will generate queries specific to data products contained within the V0 archives.

Input: Search request for inventory within V0 submitted.

Output: V0 inventory search results.

Expected Results: ECS user authenticated for V0 inventory search, user able to submit search request and receive information related to query.

Test Case 2 V0 Guide Data Search

This test verifies the tester is able to access and search V0 guide data products. The tester will generate queries specific to data products contained within the V0 archives.

Input: Search request for guide data within V0 submitted.

Output: V0 guide data search results.

Expected Results: ECS tester authenticated for V0 guide data search, tester able to submit search request and receive information related to query.

Test Case 3 V0 Browse Request

This test verifies the tester is able to access and browse V0 data products. The tester will generate queries for data and attempt to browse the data products received.

Input: Search request submitted.

Output: Search results packages.

Expected Results: ECS tester authenticated for V0 data search, tester is able to browse data products received from search request.

Test Case 4 V0 Data Product Order

This test verifies the tester is able to access V0 data products and submit a data product order. The tester will generate a query for each type of data product and submit a data order for each data type.

Input: Search request. Data product order.

Output: Search package results.

Expected Results: ECS tester authenticated for V0 data search, tester is able to browse data products and submit a data order request.

5.1.19.2.9 Sequence 9 - V0 (User)/ECS Information Management

This series of tests will verify the tester is able to interface with ECS from V0 to perform inventory and guide searches, browse requests, and order data products.

Test Case 1 ECS Inventory Search

This test verifies the tester is able to access and search ECS data products. The tester will generate queries specific to data products contained within the ECS archives.

Input: Search request for inventory within ECS submitted.

Output: ECS inventory search results.

Expected Results: V0 user authenticated for ECS inventory search, user able to submit search request and receive information related to query.

Test Case 2 ECS Guide Data Search

This test verifies the tester is able to access and search ECS guide data products. The tester will generate queries specific to data products contained within the ECS archives.

Input: Search request for guide data within ECS submitted.

Output: ECS guide data search results.

Expected Results: Tester authenticated for ECS guide data search, tester able to submit search request and receive information related to query.

Test Case 3 ECS Browse Request

This test verifies the tester is able to access and browse ECS data products. The tester will generate queries for data and attempt to browse the data products received.

Input: Search request submitted.

Output: Search results packages.

Expected Results: Tester authenticated for ECS data search, tester is able to browse data products received from search request.

Test Case 4 ECS Data Product Order

This test verifies the tester is able to access ECS data products and submit a data product order. The tester will generate a query for each type of data product and submit a data order for each data type.

Input: Search request. Data product order.

Output: Search package results.

Expected Results: Tester authenticated for ECS data search, tester is able to browse data products and submit a data order request.

5.1.19.2.10 Sequence 10 - V0/ECS Data Product Transfer

The following series of tests demonstrate the ability to transfer V0 data products to the ECS system for ingest purposes. Testing includes verification of receiving data transmission status, data store status and data checks.

Test Case 1 Transfer V0 Standard Products

This test verifies that V0 standard products are transferred to ECS ingest services upon data transmission request for V0 standard products.

Input: Request for transmission of standard products. Standard products

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 standard products are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 2 Transfer V0 Associated Correlative Data Sets

This test verifies that V0 associated correlative data sets are transferred to ECS ingest services upon data transmission request for V0 associated correlative data sets.

Input: Request for transmission of associated correlative data sets. Associated correlative data sets

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 associated correlative data sets are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 3 Transfer V0 Associated Ancillary Data Sets

This test verifies that V0 associated ancillary data sets are transferred to ECS ingest services upon data transmission request for V0 associated ancillary data sets.

Input: Request for transmission of associated ancillary data sets. Associated ancillary data sets

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 associated ancillary data sets are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 4 Transfer V0 Associated Calibration Data Sets

This test verifies that V0 associated calibration data sets are transferred to ECS ingest services upon data transmission request for V0 associated calibration data sets.

Input: Request for transmission of associated calibration data sets. Associated calibration data sets

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 associated calibration data sets are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 5 Transfer V0 Associated Metadata

This test verifies that V0 associated metadata are transferred to ECS ingest services upon data transmission request for V0 associated metadata.

Input: Request for transmission of associated metadata. Associated metadata

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 associated metadata are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 6 Transfer V0 Documentation

This test verifies that V0 documentation are transferred to ECS ingest services upon data transmission request for V0 documentation.

Input: Request for transmission of V0 documentation. V0 documentation

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 documentation is sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 7 Transfer V0 Algorithms

This test verifies that V0 algorithms are transferred to ECS ingest services upon data transmission request for V0 algorithms.

Input: Request for transmission of V0 algorithms. V0 algorithms.

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of transmission request, V0 algorithms are sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

Test Case 8 Re-Transfer of V0 Data Products

This test verifies that upon receipt of a re-transmission of a data product from ECS, due to transmission difficulties, data checks, or data storage problems, V0 will respond and re-transmit the requested data.

Input: Request for re-transmission of data products from ECS. V0 data product sent to ECS.

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of re-transmission request, V0 data product is sent to ECS ingest services. Data transmission, data checks and storage status are returned to V0 for verification of transmission.

5.1.20 NOAA Interoperability Thread (T15-04)

5.1.20.1 Thread Objectives

The objectives of the NOAA Interoperability thread are as follows:

- Provide both outgoing and incoming interoperability.
- Provide bi-directional on-line interfacing between ECS and NOAA.
- Provide ECS access to NOAA National Meteorological Center data sets for generation of ECS products.

5.1.20.2 Thread Test Description

To demonstrate two-way interoperability, ECS and NOAA must exchange directory metadata in order for each system to advertise data holdings of the other system. Updates to any metadata must be made available to all clients to fully demonstrate two-way interoperability. Log file messages should be generated indicating successful addition of metadata and successful advertisement to all clients.

Also, ECS must allow users (testers) to browse, search, and order data products from NOAA and vice versa, in order to further demonstrate two-way interoperability. The NASA Science Internet will be monitored to verify all "hand-shaking", user authentication and authorization, between systems is transparent to the user.

The tester will also verify the ability of ECS to access the NOAA National Meteorological Center (NMC) to request data sets needed as ancillary data used in the generation of ECS products. The ESN, NSI and other internet providers (where applicable), will be monitored to verify access and data transfer between ECS and NOAA.

Dependencies: (If Applicable)

In order to demonstrate the objectives of the NOAA Interoperability thread, the following interface and functional capabilities are required:

- Each system must contain directory metadata which can be manipulated to verify advertisements across systems.
- Access to functionality to manipulate the directory metadata from each system.
- Interface with NSI for access to data holdings from ECS and NOAA for search queries.
- Interface with ESN for access to NOAA data sets to support generation of ECS products.

Access to functionality within ECS to request data sets from NOAA for data production.

Test Support Requirements

- Hardware:
  - TBS
- Software:
  - TBS
- Data:
  - TBS

5.1.1.20.2.1 Sequence 1 - Directory Metadata Exchange

This sequence of tests will verify that when the directory metadata is modified (in either ECS controlled inventory or NOAA controlled inventory) that all clients, in both NOAA and ECS, are notified of the modifications. This notification will demonstrate advertisement interoperability between NOAA and ECS.

Test Case 1 ECS altered Metadata

The tester will construct a search request for ECS directory metadata from a NOAA client. The tester will submit the request, and verify the search package results. The tester will then alter the metadata, and submit the altered metadata to ECS from a NOAA client. Then the tester will re-submit the initial search request, which should result in a null match (metadata has changed from original state). The tester will then construct a search request, specific to the modifications made to the original directory metadata, and submit the request. The final search package results will verify the directory metadata has been updated.

Input: Search request, valid directory metadata update transaction. Re-submittal of original search request. Search request for updated directory metadata.

Output: Search results packages (initial and follow-up after metadata update), message log indicating directory metadata change and successful advertising to all clients of change.

Expected Results: The third search result package should reflect the modified metadata inventory.

Test Case 2 NOAA altered Metadata

The tester will construct a search request for NOAA directory metadata from an ECS client. The tester will submit the request, and verify the search package results. The tester will then alter the metadata, and submit the altered metadata to NOAA from an ECS client. Then the tester will re-submit the initial search request, which should result in a null match (metadata has changed from original state). The tester will then construct a search request, specific to the modifications made to the original directory metadata, and submit the request. The final search package results will verify the directory metadata has been updated.

Input: Search request, valid directory metadata update transaction. Re-submittal of original search request. Search request for updated directory metadata.

Output: Search results packages (initial and follow-up after metadata update), message log indicating directory metadata change and successful advertising to all clients of change.

Expected Results: The third search result package should reflect the modified metadata inventory.

5.1.20.2.2 Sequence 2 - New Data Availability

This sequence of tests will verify that when new data is added to a data site, the inventory is updated to reflect the new data. The test will verify that both NOAA and ECS are informed of updates to the existing data schemas.

Test Case 1 Addition of ECS Data

The tester will construct a simple search requesting ECS data (same type that will be added) from a NOAA client. The tester will submit the request, and verify the search package results. The tester will then proceed to ingest data into the ECS system. After the data is ingested, the tester will submit the same search request, from the NOAA client. The returned result package will reflect the newly added ECS data.

Input: Search request, data to ingest.

Output: Search package results. (initial and follow-up after ingesting data), message log indicating data ingested and successful advertising to all clients of new data.

Expected Results: The second search package result should reflect the newly ingested data.

Test Case 2 Addition of NOAA Data

The tester will construct a simple search requesting NOAA data (same type that will be added) from a ECS client. The tester will submit the request, and verify the search package results. The tester will then proceed to ingest data into the NOAA system. After

the data is ingested, the tester will submit the same search request, from the ECS client. The returned result package will reflect the newly added NOAA data.

Input:	Search request, data to ingest.
Output:	Search package results. (initial and follow-up after ingesting data), message log indicating data ingested and successful advertising to all clients of new data.
Expected Results:	The second search package result should reflect the newly ingested data.

5.1.20.2.3 Sequence 3 - Removal of Data

This sequence of tests will verify that when data is removed from a data site, that the inventory is updated to reflect the loss of data. The test will verify that both NOAA and ECS are informed of updates to the existing data schemas.

Test Case 1 Removal of ECS Data

The tester will construct a search requesting ECS data (same type that will be removed) from a NOAA client. The tester will submit the request, and verify the search package results. The tester will then proceed to remove data from the ECS system. After the data is deleted, the tester will submit the same search request, from the NOAA client. The returned package results should reflect the fact that the ECS data was deleted.

Input:	Search request, data to delete
Output:	Search package results (initial and follow-up after deleting data), message log indicating data deleted.
Expected Results:	The second search result package should reflect the deleted data.

Test Case 2 Removal of NOAA Data

The tester will construct a search requesting NOAA data (same type that will be removed) from a ECS client. The tester will submit the request, and verify the search package results. The tester will then proceed to remove data from the NOAA system. After the data is deleted, the tester will submit the same search request, from the ECS client. The returned package results should reflect the fact that the NOAA data was deleted.

Input:	Search request, data to delete
Output:	Search package results (initial and follow-up after deleting data), message log indicating data deleted.
Expected Results:	The second search result package should reflect the deleted data.

5.1.20.2.4 Sequence 4 - ECS/NOAA User Access

This sequence of tests will verify the interface between ECS and NOAA is transparent to the tester. The tester will request metadata specific to ECS from NOAA, and from ECS to NOAA to verify the user authentication process is transparent.

Test Case 1 ECS User Requesting NOAA Metadata

The tester will construct a search specific to the NOAA directory metadata from an ECS client. The tester will submit the request to ECS and select the option to continue the search at NOAA.

Input: Search request.

Output: Option to limit search within ECS system, or continue search to NOAA.

Expected Results: Search request is forwarded to NOAA for further processing. User authentication from ECS to NOAA should be transparent to the tester.

Test Case 2 NOAA User Requesting ECS Metadata

The tester will construct a simple search specific to the ECS directory metadata from a NOAA client. The tester will submit the request to NOAA and select the option to continue the search at ECS.

Input: Search request.

Output: Option to limit search within NOAA system, or continue search to ECS.

Expected Results: Search request is forwarded to ECS for further processing. User authentication from NOAA to ECS should be transparent to the tester.

5.1.20.2.5 Sequence 5 - ECS (User)/NOAA Information Management

These series of tests will verify the tester is able to interface with NOAA from ECS to perform inventory and guide searches, browse requests, and order data products.

Test Case 1 NOAA Inventory Search

This test verifies the tester is able to access and search NOAA data products. The tester will generate queries specific to data products (oceanographic, climatic, geographic) contained within the NOAA archives.

Input: Search request for inventory within NOAA pertaining to oceanographic, climatic, or geographic data submitted.

Output: NOAA inventory search results.

Expected Results: ECS user authenticated for NOAA inventory search, user able to submit search request and receive information related to query.

Test Case 2 NOAA Guide Data Search

This test verifies the tester is able to access and search NOAA guide data products. The tester will generate queries specific to data products contained within the NOAA archives.

Input: Search request for guide data within NOAA pertaining to oceanographic, climatic, or geographic data submitted.

Output: NOAA guide data search results.

Expected Results: ECS tester authenticated for NOAA guide data search, tester able to submit search request and receive information related to query.

Test Case 3 NOAA Browse Request

This test verifies the tester is able to access and browse NOAA data products. The tester will generate queries for data and attempt to browse the data products received.

Input: Search request submitted.

Output: Search results packages.

Expected Results: ECS tester authenticated for NOAA data search, tester is able to browse data products received from search request.

Test Case 4 NOAA Data Product Order

This test verifies the tester is able to access NOAA data products and submit a data product order. The tester will generate a query for each type of data product and submit a data order for each data type.

Input: Search request. Data product order.

Output: Search package results.

Expected Results: ECS tester authenticated for NOAA data search, tester is able to browse data products and submit a data order request.

Test Case 5 NOAA Data Order Status

This test verifies the tester is able to access NOAA and submit a request for the status of the previous data product order.

Input: Data order status request.

Output: Data order status.

Expected Results: Tester authenticated for NOAA system. Tester is able to submit data order status request, and receives status.

5.1.20.2.6 Sequence 6 - NOAA (User)/ECS Information Management

This series of tests will verify the tester is able to interface with ECS from NOAA to perform inventory and guide searches, browse requests, and order data products.

Test Case 1 ECS Inventory Search

This test verifies the tester is able to access and search ECS data products. The tester will generate queries specific to data products (oceanographic, climatic, geographic) contained within the ECS archives.

Input: Search request for inventory within ECS pertaining to oceanographic, climatic, or geographic data submitted.

Output: ECS inventory search results.

Expected Results: NOAA user authenticated for ECS inventory search, user able to submit search request and receive information related to query.

Test Case 2 ECS Guide Data Search

This test verifies the tester is able to access and search ECS guide data products. The tester will generate queries specific to data products contained within the ECS archives.

Input: Search request for guide data within ECS pertaining to oceanographic, climatic, or geographic data submitted.

Output: ECS guide data search results.

Expected Results: Tester authenticated for ECS guide data search, tester able to submit search request and receive information related to query.

Test Case 3 ECS Browse Request

This test verifies the tester is able to access and browse ECS data products. The tester will generate queries for data and attempt to browse the data products received.

Input: Search request submitted.

Output: Search results packages.

Expected Results: Tester authenticated for ECS data search, tester is able to browse data products received from search request.

Test Case 4 ECS Data Product Order

This test verifies the tester is able to access ECS data products and submit a data product order. The tester will generate a query for each type of data product and submit a data order for each data type.

Input: Search request. Data product order.

Output: Search package results.

Expected Results: Tester authenticated for ECS data search, tester is able to browse data products and submit a data order request.

Test Case 5 ECS Data Order Status

This test verifies the tester is able to access ECS and submit a request for the status of the previous data product order.

Input: Data order status request.

Output: Data order status.

Expected Results: Tester authenticated for ECS system. Tester is able to submit data order status request, and receives status.

5.1.20.2.7 Sequence 7 - ECS Data Product Request

The following series of tests demonstrate ECS is able to communicate with NOAA NMC through ESN, NSI, or other internet providers, to request data products used in the generation of ECS data products. The tests will also verify the data products transmitted from NOAA NMC are properly formatted, which is documented in the ADC ICD.

Test Case 1 Ancillary Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request ancillary data products.

Input: Ancillary data product request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Ancillary data request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 2 In Situ Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request in situ data products.

Input: In situ data product request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). In situ data request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 3 Algorithm Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request algorithms.

Input: Algorithm request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Algorithm request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 4 Science Engineering Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request science engineering data products.

Input: Science engineering data product request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Science engineering data request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 5 L0-L4 Equivalent Data Sets Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request L0-L4 data sets.

Input: L0-L4 data set request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). L0-L4 data set request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 6 Metadata Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request metadata.

Input: Metadata request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Metadata request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 7 Calibration Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request calibration data products.

Input: Calibration data request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Calibration data request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 8 Correlative Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request correlative data products.

Input: Correlative data request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Correlative data request is processed by NMC. If data is available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 9 Document Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request documents.

Input: Document request.

Output: Data Availability Schedule for expected delivery times, or Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Document request is processed by NMC. If available, NMC transmits a Data Availability Notification, otherwise NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 10 Management Data Request

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to request management data.

Input: Management data request.

Output: Data Availability Notification to signal availability of data.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Management data request is processed by NMC. NMC transmits a Data Availability Schedule for expected delivery times.

Test Case 11 Re-transmission Request

This test verifies that upon receipt of a re-transmission of a data product from ECS, due to transmission difficulties, data checks, or data storage problems, NOAA will respond and re-transmit the requested data.

Input: Request for re-transmission of data products from ECS. NOAA data product sent to ECS.

Output: Receive transmission status, ECS ingest data checks, and data storage status.

Expected Results: Upon receipt of re-transmission request, NOAA data product is sent to ECS ingest services. Data transmission, data checks and storage status are returned to NOAA for verification of transmission.

Test Case 12 Ancillary Data Product Format

This test verifies the ancillary data requested by ECS is delivered in the proper format, agreed upon and documented within the ADC ICD.

Input: Ancillary data product request.

Output: Data Availability Notification, data transmitted to ECS.

Expected Results: Data sets transmitted are in the proper format, which is documented within the ADC ICD.

5.1.20.2.8 Sequence 8 - Data Product Receipt

The following series of tests demonstrate ECS is able to communicate with NOAA NMC through ESN, NSI, or other internet providers, to transmit data products.

Test Case 1 L0-L4 Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send L0-L4 data.

Input: L0-L4 data transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). L0-L4 data transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 2 Metadata Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send metadata.

Input: Metadata transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Metadata transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 3 Ancillary Data Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send ancillary data.

Input: Ancillary data transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Ancillary data transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 4 Calibration Data Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send calibration data.

Input: Calibration data transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Calibration data transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 5 Correlative Data Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send correlative data.

Input: Correlative data transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Correlative data transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 6 Document Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send document information.

Input: Document transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Document transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

Test Case 7 Algorithms Transmission

This test verifies the tester is able to interface with NOAA NMC from ECS through either ESN, NSI, or other internet interface, to send algorithms.

Input: Algorithm transfer submission.

Output: Successful data transfer to NOAA system. Security log records all transactions and data transmission.

Expected Results: Tester is able to access NOAA NMC through the internet provider (NSI, ESN, etc.). Algorithm transmission request is submitted to NMC. If storage available, transmission initiated. Check sum and transmission status received at ECS.

5.1.21 FOS User Interface Thread (T14-01)

5.1.21.1 Thread Objectives

The objectives of the user interface thread are as follows:

- Provide for the verification of EOC/IST user station security functionality.
- Provide for the verification of EOC/IST user station display functionality.
- Provide for the verification of EOC/IST user station non-command PROC functionality.

5.1.21.2 Thread Test Description

The EOC user station(s) is initialized and sign-on is performed, with varying degrees of functional access types (i.e., read-only, update users, etc.). Default functionality provided at sign-on for each type of user at the EOC and IST user stations will then be verified. Default functionality includes user station header areas, status and display windows, room assignments, printer assignments, and command input areas. IST user station functionality will be performed following the verification of EOC functionality.

Dependencies:

None

Test Support Requirements

- Hardware:
  - Several EOC user stations
  - Several IST user stations
  - FOS LAN
- Data:
  - ASCII/postscript files and rastor images available at EOC and IST user stations

Sequence 1 - EOC User Authentication

The following series of tests verify those requirements related to FUI-provided user authentication capabilities, including the display of default functionality based on user privileges and disallowing unauthorized access to the EOC set of user stations.

Test Case 1 User Authentication - EOC Read Only User

This test case verifies the ability to authenticate read-only EOC users and to allow for specific sets of default user station read-only capabilities depending on defined privileges.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room/window and utility displays. Event message displays.

Expected Results: Availability of read-only functions matches the access privilege definitions based on the sign-on username/password/user type provided for EOC read-only users.

Test Case 2 User Authentication EOC Update User

This test case verifies the ability to provide security login procedures to authenticate users signing on to the EOC user stations and to allow for specific sets of default user station operability depending on defined privileges (i.e., command authority access, analysis function access, directory access, etc.).

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room and window displays, EOC user station event messages, status window.

Expected Results: Availability of functions matches the access privilege definitions based on the sign-on username/passwords provided for EOC update users.

Test Case 3 User Authentication EOC Unauthorized User

This test case verifies the ability to screen unauthorized users from accessing the EOC user station.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room and window displays.

Expected Results: No unauthorized user is allowed access to any EOC functionality; Repeated attempts to enter the system disengage username/password prompt and warning messages are displayed/logged.

5.1.21.2.2 Sequence 2 - EOC User Station Management

The following series of tests verify those requirements related to FUI-provided room and window default settings and the capabilities associated with the modification of default settings within the FUI user station environment.

Test Case 1 EOC Default Settings

This test verifies screen default display following sign-in, including command line/input areas, status window, event control window, window lists, room lists and utility option windows.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station window, room, utility and option window displays, event messages.

Expected Results: Defined default functional windows, buttons, lists displayed, accessible and functional following sign-on.

Test Case 2 EOC Window Management

This test verifies the ability to dynamically access, move, focus, iconify, reposition and resize windows within a room.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room/window displays.

Expected Results: Windows displayed are accessible, moveable, and resizable based on mouse selection criteria chosen.

Test Case 3 EOC Room Definition

This test case verifies the ability to manage the EOC user station room and window environment. Verification activities performed include accessing default rooms, defining, deleting, modifying and saving rooms. Other room functions (i.e., window management within rooms, room switching, dynamically repositioning windows within a room, and the status window display) will also be verified.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station status window, screen displays and event messages.

Expected Results: Room window management functions match those attempted via user interface input; default room/window definitions are re-applied following sign-on.

Test Case 4 EOC Room States

This test verifies the ability to define and display room states, including default, tiled and user modified room display.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room/window displays; event messages.

Expected Results: Room display state matches that specified; all display states available for any specified room. Room states can be individually selected following repositioning of windows within a room and resizing windows within a room.

Test Case 5 EOC Control Window

This test verifies user station control window functionality, including room, window, and tool option access.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room/window displays.

Expected Results: Room, window, and tool selection windows appear when selected. Multiple window/room access options may be done simultaneously.

Test Case 6 EOC User Input Recall

This test verifies user input at the command line and recall of any of the last 20 STOL user directives entered.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station room/window, event displays.

Expected Results: Upon user selection, any STOL input made, up to the 20th input, is automatically displayed in the command line area; command may be re-executed on request.

Test Case 7 EOC Print Activity

This test case verifies the ability to print various types of files at the EOC user station assigned printer(s). Capabilities tested include printing ASCII/postscript files and rastor images.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: EOC user station screen displays, event messages and printer snaps.

Expected Results: Snaps of all requested files are made at the requested EOC user station printer(s) only. All printouts contain complete information and accompanying header(s).

5.1.21.2.3 Sequence 3 - EOC Non-Command PROCs

The following series of test cases verify those requirements related to the generation, access, control and use of non-command PROCs via the FUI at the EOC set of user stations.

Test Case 1 EOC Non-Command PROC Input

This test case verifies the ability to create, edit, store, delete and print preplanned non-command PROCs at the EOC user station, using the PROC builder FUI function. Also included in this test case is the inability to use command directives when defining local PROCs.

Input: PROC builder input, STOL directives, mouse/function key input.

Output: PROC builder screen displays, event messages, STOL and STOL argument window selections.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs containing command directives are not saved and event messages produced.

Test Case 2 EOC Non-Command PROC Identifiers

This test case verifies the ability to save, list, display PROCs based on a spacecraft and/or instrument identifier.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs are saved/listed/accessible based on defined identifiers made upon save.

Test Case 3 EOC Non-Command PROC Categories

This test case verifies the ability to create, edit, save, list, display several categories of PROCs, including ground, local and user-defined PROCs.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those allowed for the specified PROC category. Event messages match editing activity. All PROCs saved are saved within specified PROC category.

Test Case 4 EOC Non-Command PROC Access

This test case verifies the ability to allow/disallow PROC access based on user sign-on information and PROC category specified during the PROC build process at the EOC user station.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, PROC directory displays, event messages.

Expected Results: PROC access functions allowed per user request match those allowed according to privileges at sign-on and specified PROC category (i.e., CERES AM-1 proc editing access not allowed by MISR IST users). Event messages match PROC access activity.

Test Case 5 EOC Non-Command Local PROC Input

This test case verifies the ability to create, edit, store, delete and print preplanned non-command PROCs at the IST user station, and the ability to select PROC input from STOL directive and STOL directive argument windows. Also included in this test case is the inability to use command directives when defining local PROCs.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages, STOL and STOL argument window selections.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs containing command directives are not saved and event messages produced.

Test Case 6 EOC User-defined PROC categories

This test case verifies the ability to create, store, delete and print PROCs within a user-defined category at the IST user station. Also included is the inability to use command directives when defining PROCs within a user defined PROC category.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs saved are saved within the user-defined PROC category. All PROCs defined within the user-specified category are not available in other PROC category directories.

Test Case 7 EOC Invalid Directives in PROCs

This test case verifies the ability to recover from an invalid directive previously entered within a non-command PROC.

Input: User directives.

Output: PROC screen displays, event messages.

Expected Results: Event message is displayed upon FUI processing of invalid directive. User able to restart PROC following invalid directive at the next directive within the PROC.

5.1.21.2.4 Sequence 4 - IST User Authentication

The following series of tests verify those requirements related to FUI-provided user authentication capabilities, including the display of default functionality based on user privileges and disallowing unauthorized access to the IST set of user stations.

Test Case 1 User Authentication - IST Read Only User

This test case verifies the ability to authenticate read-only IST users and to allow for specific sets of default user station read-only capabilities depending on defined privileges.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room/window and utility displays. Event message displays.

Expected Results: Availability of read-only functions matches the access privilege definitions based on the sign-on username/password/user type provided for IST read-only users.

Test Case 2 User Authentication - IST Update User

This test case verifies the ability to provide security login procedures to authenticate users signing on to the IST user stations and to allow for specific sets of default user station operability depending on defined privileges (i.e., command authority access, analysis function access, directory access, etc.).

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room and window displays, IST user station event messages, status window.

Expected Results: Availability of functions matches the access privilege definitions based on the sign-on username/passwords provided for IST update users.

Test Case 3 User Authentication - PI/TL User

This test case verifies the ability to provide security login procedures to authenticate users signing on to the IST user stations and to allow for specific sets of default user station operability depending on defined privileges (i.e., command authority access, analysis function access, directory access, etc.).

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room and window displays, IST user station event messages.

Expected Results: Availability of functions matches the access privilege definitions based on the sign-on username/passwords provided for each set of defined IST users.

Test Case 4 User Authentication - IST Unauthorized User

This test case verifies the ability to screen unauthorized users from accessing the IST user station.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room and window displays.

Expected Results: No unauthorized user is allowed access to any IST functionality; Repeated attempts to enter the system disengage username/password prompt and warning messages are displayed/logged.

5.1.21.2.5 Sequence 5 - IST User Station Management

The following series of tests verify those requirements related to FUI-provided room and window default settings and the capabilities associated with the modification of default settings within the FUI user station environment.

Test Case 1 IST Default Settings

This test verifies screen default display following sign-in, including command line/input areas, status window, event control window, window lists, room lists and utility option windows.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station window, room, utility and option window displays, event messages.

Expected Results: Defined default functional windows, buttons, lists displayed, accessible and functional following sign-on.

Test Case 2 IST Window Management

This test verifies the ability to dynamically access, move, focus, iconify, reposition and resize windows within a room.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room/window displays.

Expected Results: Windows displayed are accessible, moveable, and resizable based on mouse selection criteria chosen.

Test Case 3 IST Room Definition

This test case verifies the ability to manage the IST user station room and window environment. Verification activities performed include accessing default rooms, defining, deleting, modifying and saving rooms. Other room functions (i.e., window management within rooms, room switching, dynamically repositioning windows within a room, and the status window display) will also be verified.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station status window, screen displays and event messages.

Expected Results: Room window management functions match those attempted via user interface input; default room/window definitions are re-applied following sign-on.

Test Case 4 IST Room States

This test verifies the ability to define and display room states, including default, tiled and user modified room display.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room/window displays; event messages.

Expected Results: Room display state matches that specified; all display states available for any specified room. Room states can be individually selected following repositioning of windows within a room and resizing windows within a room.

Test Case 5 IST Control Window

This test verifies user station control window functionality, including room, window, and tool option access.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room/window displays.

Expected Results: Room, window, and tool selection windows appear when selected. Multiple window/room access options may be done simultaneously.

Test Case 6 IST User Input Recall

This test verifies user input at the command line and recall of any of the last 20 STOL user directives entered.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station room/window, event displays.

Expected Results: Upon user selection, any STOL input made, up to the 20th input, is automatically displayed in the command line area; command may be re-executed on request.

Test Case 7 IST Print Activity

This test case verifies the ability to print various types of files at the IST user station assigned printer(s). Capabilities tested include printing ASCII/postscript files and rastor images.

Input: User directives, other user interface input (i.e., function keys, mouse movements, mouse clicks, etc.).

Output: IST user station screen displays, event messages and printer snaps.

Expected Results: Snaps of all requested files are made at the requested IST user station printer(s) only. All printouts contain complete information and accompanying header(s).

5.1.21.2.6 Sequence 6 - IST Non-Command PROCs

The following series of test cases verify those requirements related to the generation, access, control and use of non-command PROCs via the FUI at the IST set of user stations.

Test Case 1 IST Non-Command PROC Input

This test case verifies the ability to create, edit, store, delete and print preplanned non-command PROCs at the IST user station, using the PROC builder FUI function. Also included in this test case is the inability to use command directives when defining local PROCs.

Input: PROC builder input, STOL directives, mouse/function key input.

Output: PROC builder screen displays, event messages, STOL and STOL argument window selections.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs containing command directives are not saved and event messages produced.

Test Case 2 IST Non-Command PROC Identifiers

This test case verifies the ability to save, list, display PROCs based on a spacecraft and/or instrument identifier.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs are saved/listed/accessible based on defined identifiers made upon save.

Test Case 3 IST Non-Command PROC Categories

This test case verifies the ability to create, edit, save, list, display several categories of PROCs, including ground, local and user-defined PROCs.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those allowed for the specified PROC category. Event messages match editing activity. All PROCs saved are saved within specified PROC category.

Test Case 4 IST Non-Command PROC Access

This test case verifies the ability to allow/disallow PROC access based on user sign-on information and PROC category specified during the PROC build process at the IST user station.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, PROC directory displays, event messages.

Expected Results: PROC access functions allowed per user request match those allowed according to privileges at sign-on and specified PROC category (i.e., CERES AM-1 proc editing access not allowed by MISR IST users). Event messages match PROC access activity.

Test Case 5 IST Non-Command Local PROC Input

This test case verifies the ability to create, edit, store, delete and print preplanned non-command PROCs at the IST user station, and the ability to select PROC input from STOL directive and STOL directive argument windows. Also included in this test case is the inability to use command directives when defining local PROCs.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages, STOL and STOL argument window selections.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs containing command directives are not saved and event messages produced.

Test Case 6 IST User-defined PROC categories

This test case verifies the ability to create, store, delete and print PROCs within a user-defined category at the IST user station. Also included is the inability to use command directives when defining PROCs within a user defined PROC category.

Input: STOL directives, mouse, function keys.

Output: PROC screen displays, event messages.

Expected Results: PROC editing functions allowed per user request match those attempted. Event messages match editing activity. All PROCs saved are saved within the user-defined PROC category. All PROCs defined within the user-specified category are not available in other PROC category directories.

Test Case 7 IST Invalid Directives in PROCs

This test case verifies the ability to recover from an invalid directive previously entered within a non-command PROC.

Input: User directives.

Output: PROC screen displays, event messages.

Expected Results: Event message is displayed upon FUI processing of invalid directive. User able to restart PROC following invalid directive at the next directive within the PROC.

5.1.22 FOS Resource Management Thread (T14-02)

5.1.22.1 Thread Objectives

The objectives of the resource management thread are as follows:

- Provide for the verification of FOS operational and backup configuration initialization.
- Provide for the verification of EOC/IST status display of the FOS operational/backup configurations.
- Provide for the verification of the initialization, management and reconfiguration of FOS logical strings.

- Provide for the verification of FOS external interface connectivity, including ECOM, EDOS, FDF, NCC and the ASTER IP-ICC.

5.1.22.2 Thread Test Description

The FOS (including data servers, user stations, local area networks and peripherals) is initialized to support various operational/non-operational configurations. Following FOS initialization, user interface display of configuration status including status event messages, equipment status windows and configuration display windows will be verified to ensure that the most current configuration information is available to the FOS user. FOS connectivity to its external interfaces will then be verified, along with the capability of initializing, managing and reconfiguring FOS logical strings (i.e., real-time telemetry, replay telemetry, etc.). Test steps will also be provided to ensure that configuration/reconfiguration activity is screened from unauthorized users, and that configuration/reconfiguration of any FOS element which violates operational constraints is not permitted.

Dependencies:

In order to verify the objectives of the resource management thread, the following interface and functional capabilities are required:

- User Interface
- Telemetry Thread
- Data Management Thread
- Analysis Thread

Test Support Requirements

- Hardware:
  - FOS real-time server
  - FOS data server
  - FOS data archive
  - EOC user stations
  - IST user stations
  - FOS LAN
- Data:
  - External interface test messages

5.1.22.2.1 Sequence 1 - FOS Initialization

The following series of tests verify those requirements related to the initialization, management, display and termination of the various FOS configurations needed for ground support and testing of the AM-1 spacecraft and its complement of instruments.

Test Case 1 AM-1 Operational Configuration Initialization

This test case verifies the ability to initialize the AM-1 operational configuration from the EOC and to display configuration status information at EOC and IST user stations.

Input: User directives to configure the operational string at the EOC, display status windows, and perform printer snaps. Command controller/master control designated for real-time user stations; other EOC/IST user stations configured in various modes, including scheduling, analysis and real-time modes.

Output: User interface configuration status display windows and printer snaps at EOC and IST user stations; event messages reporting equipment status at initialization and at each configuration/reconfiguration iteration.

Expected Results: Configuration information available at EOC and IST user stations accurately portray the correct AM-1 operational configuration at initialization.

Test Case 2 Simulation/test/development Configuration Initialization

This test case verifies the ability to initialize the AM-1 backup (simulation/test/development) configuration from the EOC and to display configuration status information at EOC and IST user stations.

Input: User directives to configure the simulation/test/development configuration at the EOC, display status windows, and perform printer snaps. Command controller/master control designated for real-time user stations; other EOC/IST user stations configured in various modes, including scheduling, analysis and real-time modes.

Output: User interface configuration status display windows and printer snaps at EOC and IST user stations; event messages reporting equipment status at initialization.

Expected Results: Configuration information available at EOC and IST user stations accurately portray the correct AM-1 simulation/test/development configuration at initialization.

5.1.22.2.2 Sequence 2 - FOS External Interface Connectivity

The following series of tests verify those requirements related to the initialization and test message transmission to/from FOS external interfaces, including EDOS, ECOM, NCC and the FDF.

Test Case 1 ECOM Connectivity

This test case verifies the external interface connectivity between the FOS and ECOM, and the ability to connect to ECOM in the operational configuration or backup (simulation/test/development configuration.

Input:	User directives to configure connectivity, send test messages and display ECOM status display window.
Output:	Event messages, ECOM status display windows.
Expected Results:	EOC/IST configuration display information accurately portrays the ECOM interface status; ECOM connectivity allowed for either configuration and status event messages/ECOM status window portrays the most current connectivity status. Test messages processed by ECOM/FOS.

Test Case 2 EDOS Connectivity

This test case verifies the external interface connectivity between the FOS and the EDOS, and the ability to connect to EDOS in the operational configuration or backup (simulation/test/development) configuration.

Input:	User directives to configure connectivity, send test messages and display EDOS status display window.
Output:	Event messages, EDOS status display windows.
Expected Results:	EOC/IST configuration display information accurately portrays the EDOS interface status; EDOS connectivity allowed for either configuration and status event messages/EDOS status window portrays the most current connectivity status. Test messages processed by EDOS/FOS.

Test Case 3 FDF Connectivity

This test case verifies the external interface connectivity between the FOS and the FDF, and the ability to connect to the FDF in the operational configuration or backup (simulation/test/development) configuration.

Input:	User directives to configure connectivity, send test messages, and display NCC status display window.
Output:	Event messages, FDF status display windows.
Expected Results:	EOC/IST configuration display information accurately portrays the FDF interface status; FDF connectivity allowed for either configuration and status event messages/FDF status window portrays the most current connectivity status. Test messages processed by FDF/FOS.

Test Case 4 NCC Connectivity

This test case verifies the external interface connectivity between the FOS and the NCC, and the ability to connect to the NCC in the operational configuration or backup (simulation/test/development) configuration.

Input:	User directives to configure connectivity, send test messages and display NCC status display window.
Output:	Event messages, NCC status display windows.

Expected Results: EOC/IST configuration display information accurately portrays the NCC interface status; NCC connectivity allowed for either configuration and status event messages/NCC status window portrays the most current connectivity status. Test messages processed by NCC/FOS.

Test Case 5 ASTER IP-ICC Connectivity

This test case verifies the external interface connectivity between the FOS and the ASTER IP-ICC, and the ability to connect to the ASTER IP-ICC in the operational configuration or backup (simulation/test/development) configuration.

Input: User directives to configure connectivity, send test messages and display ASTER IP-ICC status display window.

Output: Event messages, ASTER IP-ICC status display windows.

Expected Results: EOC/IST configuration display information accurately portrays the ASTER IP-ICC interface status; ASTER IP-ICC connectivity allowed for either configuration and status event messages/ASTER IP-ICC status window portrays the most current connectivity status. Test messages processed by NCC/ASTER IP-ICC.

5.1.22.2.3 Sequence 3 - FOS String Configuration

The following series of tests verify those requirements related to the initialization, management, display and termination of the various FOS string configurations needed for ground support of the AM-1 spacecraft and its complement of instruments.

Test Case 1 Initialize Real-Time Telemetry String Configuration

This test case verifies the ability to initialize the real-time telemetry string configuration, automatically configure FOS resources for the configuration, and report on the resources which provide support to the configuration via status event messages and/or equipment status windows.

Input: User station directives to initialize the real-time telemetry string.

Output: EOC user station event messages/status window display snaps.

Expected Results: Real-time telemetry string configuration is initialized with proper FOS equipment specified via the user directive(s). Status windows show FOS equipment/resources used in the telemetry string configuration. String configuration status messages sent to all configured user stations.

Test Case 2 Real-Time Telemetry String Configuration - EOC/IST User Station Connect/Disconnect

This test case verifies the ability to connect/disconnect EOC and IST user stations to/from the real-time telemetry string configuration via user directives. This test also verifies the ability to report EOC/IST user station connection/disconnection status via event messages.

Input:	User directives for configuring the real-time telemetry string configuration at EOC and IST user stations.
Output:	EOC/IST user station event messages/status window display snaps.
Expected Results:	EOC/IST user stations are able to connect/disconnect to the real-time telemetry string configuration; status event messages are generated stating connection/disconnection activity; equipment status windows are updated to show user station connect/disconnect activity.

Test Case 3 — Real-Time Telemetry String Configuration Termination

This test case verifies the ability to terminate the real-time telemetry string configuration via user directive, and to report on the termination of the real-time telemetry string.

Input:	User directives for terminating the real-time telemetry string configuration at the EOC.
Output:	EOC user station event messages/status window display snaps.
Expected Results:	Real-time telemetry string configuration is terminated resulting in event messages stating the termination; equipment status windows updated to show the changes; EOC/IST user stations currently connected to the real-time telemetry string configuration are notified of the string termination via event messages.

Test Case 4 — Initialize Telemetry Replay String Configuration

This test case verifies the ability to initialize the telemetry replay string configuration, automatically configure FOS resources for the configuration, and report on the resources which provide support to the configuration via status event messages and/or equipment status windows.

Input:	User station directives to initialize the telemetry replay string.
Output:	EOC user station event messages/status window display snaps.
Expected Results:	Telemetry replay string configuration is initialized with proper FOS equipment specified via the user directive(s). Status windows show FOS equipment/resources used in the telemetry replay string configuration. String configuration status messages sent to all configured user stations.

Test Case 5 — Telemetry Replay String Configuration - EOC/IST User Station Connect/Disconnect

This test case verifies the ability to connect/disconnect EOC and IST user stations to/from the telemetry replay string configuration via user directives. This test also verifies the ability to report EOC/IST user station connection/disconnection status via event messages.

Input: User directives for configuring the telemetry replay string configuration at EOC and IST user stations.

Output: EOC/IST user station event messages/status window display snaps.

Expected Results: EOC/IST user stations are able to connect/disconnect to the telemetry replay string configuration; status event messages are generated stating connection/disconnection activity; equipment status windows are updated to show user station connect/disconnect activity.

Test Case 6 Telemetry Replay String Configuration Termination

This test case verifies the ability to terminate the telemetry replay string configuration via user directive, and to report on the termination of the configuration via event messages.

Input: User directives for terminating the telemetry replay string configuration at the EOC.

Output: EOC user station event messages/status window display snaps.

Expected Results: Telemetry replay string configuration is terminated resulting in event messages stating the termination; equipment status windows updated to show the changes; EOC/IST user stations currently connected to the telemetry replay string configuration are notified of the string termination via event messages.

Test Case 7 Initialize Simulation/Test/Development Replay String Configuration

This test case verifies the ability to initialize the simulation/test/development configuration, automatically configure FOS resources for the configuration, and report on the resources which provide support to the configuration via status event messages and/or equipment status windows.

Input: User station directives to initialize the simulation/test/development string.

Output: EOC user station event messages/status window display snaps.

Expected Results: Simulation/test/development string configuration is initialized with proper FOS equipment specified via the user directive(s). Status windows show FOS equipment/resources used in the simulation/test/development string configuration. String configuration status messages sent to all configured user stations.

Test Case 8 Simulation/test/development String Configuration - EOC/IST User Station Connect/Disconnect

This test case verifies the ability to connect/disconnect EOC and IST user stations to/from the simulation/test/development string configuration via user directives. This test also verifies the ability to report EOC/IST user station connection/disconnection status via event messages.

Input: User directives for configuring the simulation/test/development string configuration at EOC and IST user stations.

Output: EOC/IST user station event messages/status window display snaps.

Expected Results: EOC/IST user stations are able to connect/disconnect to the simulation/test/development string configuration; status event messages are generated stating connection/disconnection activity; equipment status windows are updated to show user station connect/disconnect activity.

Test Case 9 Simulation/test/development String Configuration Termination

This test case verifies the ability to terminate the simulation/test/development string configuration via user directive, and to report on the termination of the configuration via event messages.

Input: User directives for terminating the simulation/test/development string configuration at the EOC.

Output: EOC user station event messages/status window display snaps.

Expected Results: Simulation/test/development string configuration is terminated resulting in event messages stating the termination; equipment status windows updated to show the changes; EOC/IST user stations currently connected to the simulation/test/development string configuration are notified of the string termination via event messages.

Test Case 10 Unauthorized/Illegal String Configuration Requests

This test case verifies the ability to screen unauthorized users from FOS string configuration activity, and to disallow string configuration requests that cause equipment or other operational conflicts.

Input: EOC/IST user directives for configuring telemetry, telemetry replay and simulation/test/development strings.

Output: EOC/IST user station event messages.

Expected Results: String requests are allowed from users possessing master authority only. No string configuration attempts made at the IST are accepted by the EOC. Any string configuration request which results in equipment or operation constraint is not allowed and an event message stating the error is generated and displayed at the requesting user station.

5.1.22.2.4 Sequence 4 - FOS String Reconfiguration

This thread verifies requirements related to single and multi-string reconfigurations within the FOS.

Test Case 1 AM-1 Operational Reconfiguration

This test case verifies the ability to reconfigure strings in the AM-1 operational configuration (i.e., real-time telemetry string, replay telemetry string, simulation/test/development string), and to accurately report reconfiguration status messages and status displays at EOC and IST user stations.

Input: User directives to reconfigure the operational string at the EOC, display status windows, and perform printer snaps. Command controller/master control designated for real-time user stations; other EOC/IST user stations configured in various modes, including scheduling, analysis and real-time modes.

Output: User interface configuration status display windows and printer snaps at EOC and IST user stations; event messages reporting equipment status at initialization and at each configuration/ reconfiguration iteration.

Expected Results: Configuration information available at EOC and IST user stations accurately portray the correct AM-1 operational configuration at initialization and following each configuration/reconfiguration change performed.

Test Case 2 Simulation/Test/Development Reconfiguration

This test case verifies the ability to reconfigure strings in the AM-1 simulation/test/development configuration (i.e., real-time telemetry string, replay telemetry string, simulation/test/development string), and to accurately report reconfiguration status messages and status displays at EOC and IST user stations.

Input: User directives to reconfigure the operational string at the EOC, display status windows, and perform printer snaps. Command controller/master control designated for real-time user stations; other EOC/IST user stations configured in various modes, including scheduling, analysis and real-time modes.

Output: User interface configuration status display windows and printer snaps at EOC and IST user stations; event messages reporting equipment status at initialization and at each configuration/ reconfiguration iteration.

Expected Results: Configuration information available at EOC and IST user stations accurately portray the correct AM-1 simulation/test/development configuration at initialization and following each configuration/ reconfiguration change performed.

5.1.23 FOS Data Management Thread (T14-03)

5.1.23.1 Thread Objectives

The objectives of the data management thread are as follows:

- Provide FOS ingest capabilities for MMC-provided PDB files.

- Provide PDB validation and reporting capabilities.
- Provide PDB editing, copying and backup capabilities.
- Provide FOS event message broadcast, display and archive capabilities.

5.1.23.2 Thread Test Description

Following sign-on to the AM-1 spacecraft configuration and entry into the appropriate account, the transfer and ingestion of PDB files is initiated. Following PDB acceptance, the validation of the PDB files is verified and the number and types of database errors are tabulated. Output of the induced database errors and type descriptions will be verified against scripted data errors and listings of the scripted errors. Induced errors will be reported by initiating the display/print of the data base validation log and other data base report information available. Following PDB validation activity, the ability to modify and copy PDB files (both validated and non-validated PDBs) will be performed.

Following PDB validation, generation, and editing activity, an event driver will be initialized, resulting in the broadcast of event message traffic onto the FOS local area network. Event messages representing differing event types (command, telemetry, tell-all, configuration, NCC, FDF, etc.) and configurations (operational, test, etc.) will be transmitted in order to verify the display, archive and management of event messages at the EOC and IST user stations.

Dependencies:

In order to verify the objectives of the data management thread, the following interface and functional capabilities are required:

- User Interface
- Data Base Build

Test Support Requirements

- Hardware:
  - DBA Admin Terminal
  - Several EOC user stations
  - DBA Admin Terminal
  - T&C Processor
  - DBA Admin Terminal
  - FOS LAN
- Software:
  - Oracle (Data Base)
  - Event Message Generator
- Data:

- PDB with scripted errors in specified command, telemetry, activity and constraint definition records.
- PDB with no errors in command, telemetry, activity, and constraint definition records.
- Event messages covering various spacecraft, configurations, event types and sequential UTC times.

5.1.23.2.1 Sequence 1 - Accept/Validate PDB

The following series of tests verify the ability of accepting PDB files sent from the MMC, and validating the PDB files following acceptance by the FOS.

Test Case 1 — Accept/Validate PDB from MMC (no errors)

This test case verifies the ability to accept the PDB (including spacecraft and instrument files) from the MMC.

Input: MMC provided AM-1 PDB. User station directives to ingest the PDB at the EOC.

Output: Event message reporting PDB file acceptance; temporary PDB file directory, PDB file dumps.

Expected Results: Temporary PDB storage area is populated with the complete set of AM-1 PDB files; No missing files/records in temporary storage area; PDB file directory complete. PDB validation process automatically begins.

Test Case 2 — Accept/Validate PDB from MMC (scripted errors included)

This test case verifies the ability to accept the PDB (including spacecraft and instrument files) from the MMC even though certain PDB records contain scripted errors.

Input: MMC provided AM-1 PDB with scripted errors. User station directives to ingest the PDB at the EOC.

Output: Event message reporting PDB file acceptance; temporary PDB file directory, PDB file dumps.

Expected Results: Temporary PDB storage area is populated with the new complete set of AM-1 PDB files; all incoming valid files become latest versions. Older version PDB files are not automatically deleted. PDB validation process automatically begins. PDB file directory contains all PDB files sent.

5.1.23.2.2 Sequence 2 - Edit PDB

The following series of tests verify PDB browse/search and editing functions allowed for both non-validated and validated PDB files.

Test Case 1 Edit non-validated PDB

This test case verifies the ability to browse/search through non-validated PDB files and edit (modify, add, delete) any portion of the non-validated set of PDB files/records at appropriate FOS element locations.

Input: Non-validated AM-1 PDB. User station directives to browse/edit the PDB at the EOC.

Output: Non-validated PDB file directory, PDB file dumps; event messages reporting PDB browse/edit activity.

Expected Results: Non-validated PDB files/records may be browsed/searched, files may be edited, added and/or deleted. Edited files/records may be saved to the non-validated PDB directory. All edited PDB files become latest file versions in the non-validated PDB directory upon save.

Test Case 2 Edit Validated PDB

This test case verifies the ability to browse/search through validated PDB files/records, and block the user from editing, adding, or deleting any validated PDB file/record.

Input: Validated AM-1 PDB. User station directives to browse/edit the PDB.

Output: PDB file directory, PDB file dumps; event messages reporting PDB browse/edit activity.

Expected Results: Validated PDB files may be browsed only; any attempt to modify, add and/or delete files is denied.

Test Case 3 Database file/record hardcopy/printout

This test case verifies the ability to print any PDB file/record, groups of files/records, and PDB directories at hardcopy and printer devices.

Input: Validated/non-validated AM-1 PDB files. User station directives to hardcopy/print PDB and PDB directory content.

Output: Hardcopy, printer output, event messages reporting PDB hardcopy/print activity.

Expected Results: Specified PDB files/records, groups of files/records, and PDB directories are printed at specified output devices upon user directive. Output content matches the dump of the requested files/records or groups of files/records.

Test Case 4 PDB Browse

This test case verifies the ability to browse/search through operational PDB files and deny any edit activity of those PDB files at the EOC and IST user stations.

Input: User station directives to browse/edit the PDB at EOC and IST user stations.

Output: PDB file display at the EOC and IST user stations; file dumps; event messages reporting PDB browse/edit activity.

Expected Results: All operational PDB files/records located at the EOC/IST may be browsed/searched. Upon PDB file/record edit attempt, event message is reported and states denial of PDB edit activity.

Test Case 5 PDB Editing- Unauthorized Users

This test verifies the ability to screen unauthorized users from PDB editing activity, including PDB edit, copy, delete, modify, and backup.

Input: User directives to browse/edit the PDB at EOC/IST user stations; varying degree of user authorization.

Output: Event messages at EOC/IST user stations.

Expected Results: Users with read-only or any non-editing privileges are denied all PDB editing activity at the EOC/IST user stations; event messages stating unauthorized activity.

5.1.23.2.3 Sequence 3 - Operational PDB Backup

The following sequence of tests verifies the ability to produce a backup copy of the entire PDB content or specified areas of the PDB and store the content into a dedicated storage area at the EOC.

Test Case 1 Generate PDB Backup

This test case verifies the ability to create and store a PDB backup which includes all PDB files.

Input: AM-1 operational PDB. User station directives to create and store the operational PDB at the EOC.

Output: Event messages reporting operational PDB backup file copy activity, operational PDB backup directory listings; DBA user interface displays.

Expected Results: Operational PDB backup storage area is populated with the complete set of AM-1 operational backup PDB files; No missing files/records in storage area; Operational PDB backup file directory complete and matches those files sent.

Test Case 2 Generate Partial PDB Backup

This test case verifies the ability to create and store a PDB backup which includes only specified Operational PDB files.

Input: AM-1 operational PDB. User station directives to create and store specified operational PDB backup files at the EOC.

Output: Event messages reporting operational PDB backup file copy activity, operational PDB backup directory listings; DBA user interface displays.

Expected Results: Operational PDB backup storage area is populated with the specified AM-1 operational backup PDB files; Only those files specified for backup are copied to the backup area.

5.1.23.2.4 Sequence 4 - Event Message Display

The following tests verify event message display at the EOC and IST user stations.

Test Case 1 Event Message Display - All Configurations

This test verifies that all broadcast event message types defined for the AM-1 spacecraft configuration are displayed at the EOC and IST user stations. This test also verifies that event messages defined for another configuration are not displayed unless requested by the user.

Input: Event message driver simulating event message traffic on the operational ECS local area network, with all data base defined event types and sequential event UTC stamped times supported. User directives to set up event processing at the EOC and IST user stations.

Output: Event display pages at EOC and IST user stations.

Expected Results: All event messages for the AM-1 spacecraft output via the event driver displayed at the EOC and IST user stations signed on to the AM-1 configuration. Other configuration event messages displayed upon user request only and match those output by the event driver.

Test Case 2 Event Message Display - Local Event Messages

This test verifies that local event messages (i.e., user station display and print messages, other localized event messages) are displayed at the initiating EOC/IST user station only.

Input: Event message driver simulating event message traffic on the operational ECS local area network, with all data base defined event types and sequential event UTC stamped times supported. User directives to set up event processing at the EOC and IST user stations.

Output: Event message displays at the EOC/IST user stations.

Expected Results: All non-local event messages output via the event driver are displayed at initiating user stations only. Spacecraft configuration does not affect local event message processing.

Test Case 3 Event Message Display - Requesting Event Message Types

This test verifies the ability to request specific event types (i.e., NCC, FDF, command, telemetry, etc.) for display at the EOC/IST user stations.

Input: Event message driver simulating event message traffic on the operational ECS local area network, with all data base defined event types and sequential event UTC stamped times supported. User directives to set up event type display at the EOC and IST user stations.

Output: Event message displays at the EOC/IST user stations.

Expected Results: Only event message types requested are displayed at the requesting EOC/IST user station and event messages displayed match those sent by the event message driver. Requests made at any EOC/IST user station in no way affect event processing at other user stations.

Test Case 4 Event Types - Error Conditions

This test verifies the ability to report on incorrectly entered user directives at EOC/IST user stations.

Input: User directives at EOC/IST user stations.

Output: Event message displays at the EOC/IST user stations.

Expected Results: Upon directive error, event message is output stating type of error.

5.1.23.2.5 Sequence 5 - Event Message Archiving

The following tests verify the ability to archive event messages of various types and configurations, and manage the event archive at EOC and IST user stations.

Test Case 1 Event Message Archiving - All Messages

This test verifies that all broadcast event message types defined for the AM-1 spacecraft, including external interface messages, are automatically stored and cataloged in the event history archive.

Input: Event message driver simulating event message traffic on the operational ECS local area network, with all data base defined event types and sequential event UTC stamped times supported.

Output: Event archive file dumps, event display pages at EOC and IST user stations.

Expected Results: All event messages output via the event driver archived in the FOS event archive (including IST generated messages) in UTC time order received. Event archive directory files contain proper UTC and configuration naming conventions.

Test Case 2 Event Message Archive Screening - Local Event Messages

This test verifies that local event messages (i.e., user station display and print messages, other localized event messages) are not archived in the 7-day event archive.

Input: Event message driver simulating event message traffic; several user stations signed on to several configurations; user directives performed resulting in local event message output.

Output: Event archive file dumps for time periods covering user directive requests made at the various user stations.

Expected Results: All non-local event messages output via the event driver are archived in the 7-day event archive in UTC time order received; all localized event messages non-existent in the archive.

Test Case 3 Event Archive Access

This test verifies the ability to access the event archive file directory at the EOC and IST user stations.

Input: User directives to request event archive activity.

Output: Event archive display; event messages.

Expected Results: Access to event archive browse capabilities (i.e., archive file directory display) provided to authorized users only.

Test Case 4 Event Archive File Deletion

This test verifies the ability to delete event archive files from the 7-day event archive by authorized users.

Input: User directives to request archive file deletion activity.

Output: Event archive directory display.

Expected Results: File(s) deleted per user command; only master/command allowed to delete files, all others denied. Event messages reporting file deletion.

Test Case 5 Turn Event Archive Process On/Off

This test verifies the ability to turn the event archive process on and off at the EOC master user station.

Input: User directives to request archive on/off activity.

Output: Event message at EOC and IST user stations; event archive content; event archive directory.

Expected Results: Event messages stating archive process status following archive on/off directives. Event archive content following on/off directives coincides with the archive status following each on/off period. Archive processing denied to all non-authorized EOC/IST users.

5.1.24 FOS Planning and Scheduling Thread (T14-04)

5.1.24.1 Thread Objectives

The objective of this thread is to identify the Planning & Scheduling (P&S) functionality required to support mission scheduling of AM-1 spacecraft subsystem and science instrument activities. Planning & Scheduling functionality identified in TRMM (Rel A) includes the EOC capability to ingest instrument and subsystem activity information from CSMS, the instrument PI/TL ISTs, spacecraft subsystem engineers, the Flight Dynamics Facility (FDF) and the Network Control Center (NCC). The planning information received from these facilities provide the input necessary to achieve the P&S subsystem final product, a conflict-free Detailed Activity Schedule (DAS) that is made available to the Command Management Subsystem and becomes the basis for scheduled AM-1 command generation.

Specifically, the P&S thread identifies the following TRMM (Rel A) functionality related to the ingest, distribution and merger of planning & scheduling information:

- Ingest of the Long Term Science Plan (LTSP) and the Long Term Instrument Plan (LTIP) from the CSMS System Monitoring and Coordination (SMC) element.
- Ingest of Planning Data from the Goddard Space Flight Center (GSFC) Flight Dynamics Facility (FDF).
- Distribution of FDF Planning Data to Authorized PI/TLs and Spacecraft Subsystem Engineers.
- Generation of the Mission Long Term Spacecraft Operations Plan.
- Generation of Instrument Baseline Activity Profile (BAP) and BAP Deviation Lists.
- Generation of Instrument Resource Profiles & Instrument Activity Profiles.
- Generation of Spacecraft Subsystem Resource Profiles and Resource Deviations.
- EOC FOS - ASTER Instrument Control Center (ICC)/IST Connectivity.
- EOC FOS - GSFC Network Control Center (NCC) Connectivity.
- Scheduling Requests for Tracking & Data Relay Satellite System (TDRSS) .
- Generation of Preliminary Resource Schedule.
- Generation of the Planning & Scheduling Detailed Activity Schedule (DAS).
- Instrument and Spacecraft Subsystem Late Change Updates to the DAS.

5.1.24.2 Thread Test Description

P&S thread testing verifies TRMM (Rel A) functionality that provides the capability of the EOC to ingest planning & scheduling information from various instrument teams, spacecraft subsystem engineers and support facilities. Once ingested by the EOC, the planning & scheduling information is merged and/or utilized in such a way that the EOC is capable of producing a Detailed Activity Schedule. The Detailed Activity Schedule generated will provide detail of the spacecraft commanding and ground directives that the flight operations team will need to execute during upcoming spacecraft contact.

Thread testing is initiated by verifying from a flight operations view point the EOC capability to support the Long-term planning phase of the scheduling cycle. Testing verifies the (P&S) subsystem ingest of the Long-Term Science Plan (LTSP) and Long-Term Instrument Plan (LTIP) from the CSMS SMC subsystem.

Following successful LTSP and LTIP ingest testing, the test conductor will proceed to verify the EOC capability to interface with the GSFC FDF, ingest planning data from the GSFC FDF and distribute the planning data to the authorized PI/TL ISTs and EOC subsystem engineering users.

Following successful FDF interface testing, the test conductor will proceed to verify the EOC capability to support generation of the Mission Long Term Spacecraft Operations Plan. Once generated, testing will verify the capability of the EOC to support distribution of the Long Term Spacecraft Plan to authorized PI/TL ISTs and EOC users.

Following successful generation of the Spacecraft Operations Plan, the test conductor will proceed to verify the EOC capability to support the generation of instrument Baseline Activity Profiles (BAP) at the associated PI/TL/ISTs. Once the BAPs have been generated, thread testing will verify the capability to transfer the BAPs to the EOC Scheduler for input into the P&S DAS. Thread testing during this phase will also verify the capability of the EOC to support the generation of BAP deviations. Once generated, the BAP deviations are transferred to the EOC Scheduler for incorporation with the original BAP information.

Following the successful completion of BAP testing, the test conductor will proceed to verify the EOC capability to support the generation of Instrument Resource Profile (IRP) and Instrument Activity Profiles (IAP). These profiles are generated at the EOC and are based on BAP and BAP deviations as input. Testing verifies that upon successful generation of these profiles, authorized users are able to access the contents of the profiles for further instrument and subsystem scheduling.

Following successful completion of IRP and IAP testing, the test conductor will proceed to verify the EOC capability to support the generation of Spacecraft Subsystem Resource Profiles and Resource Deviations. Testing verifies that Subsystem Engineers are provided the capability to generate resource profiles and transfer the profiles to the EOC Scheduler for incorporation into the P&S DAS. Testing will also verify that Subsystem Engineers are provided the capability to generate Resource Deviations Lists that can also be transferred to the EOC Scheduler for DAS incorporation.

Following the successful completion of Subsystem Resource Profile testing, the test conductor will proceed to verify the EOC capability to support EOC FOS - ASTER ICC connectivity. Due to the limited functionality of this interface in TRMM (Rel A), testing will consist primarily of "handshakes" across the interface. Testing will include test messages and acknowledgments transferred across the interface to confirm connectivity only.

Following EOC-ASTER ICC connectivity testing, the test conductor will proceed to verify the EOC capability to support EOC-GSFC NCC connectivity and the transfer of TDRSS schedule request across the interface. Testing will verify the capability of the EOC to transfer TDRSS schedule requests to the NCC and the capability of the EOC to receive schedule request approvals and rejections from the NCC.

Following EOC-NCC interface testing, the test conductor will proceed to verify the EOC capability to support the generation of Preliminary Resource Schedules (PRS). Preliminary Resource Schedules are generated at the EOC based on Instrument Resource Profiles, Subsystem Resource Profiles and accepted TDRSS schedule request. Upon successful generation of the PRS, testing will verify the capability of the EOC to support authorized user access to the contents of the PRS.

Planning & Scheduling Thread testing will conclude with the verification of the EOC capability to support generation of a Detailed Activity Schedule based on the activity and resource inputs received from the various project teams and support facilities. Upon generation of the DAS, testing will verify the capability to support constraint checking of the DAS, the capability to update the DAS based on Late Changes and Targets of Opportunity (TOOs) and transfer of the DAS to a directory accessible by the Command Management Subsystem for Absolute Time Command and Flight Operations Ground Script generation (ATC and Ground Script generation tested as part of the Command Management Thread).

Dependencies: (If Applicable)

In order to verify the objectives of the planning and scheduling thread, the following interface and functional capabilities are required:

- FOS User Interface Thread
- EOC - PI/TL IST Interface
- EOC - FDF Interface
- EOC - ASTER ICC/IST Interface
- EOC - GSFC NCC Interface
- EOC FOS - CSMS SMC Interface

Test Support Requirements:

- Hardware:
  - TBS
- Software:
  - TBS
- Data:
  - TBS

5.1.24.2.1 Sequence 1 - Long-Term Planning Ingest and Distribution

The following series of tests verify the Planning & Scheduling (P&S) subsystem's capability to support the ingest of a Long Term Science Plan (LTSP) and a Long Term Instrument Plan (LTIP) from the CSMS SMC. In addition, this sequence of test cases verifies the capability of the EOC to ingest planning data from the FDF and distribute the planning data to authorized EOC and IST users. The final phase of this sequence will verify the EOC capability to support

generation of the Long Term Spacecraft Operations Plan and distribution of the Plan to authorized users.

Test Case 1 P&S Subsystem Ingest of the Long-Term Science Plan (LTSP)

This test verifies the P&S capability to ingest the Long Term Science Plan (LTSP) from the CSMS SMC. Upon receipt and store, the contents of the plan are verified.

Input: Coordination/Input at the Planner/Scheduler workstation to invoke the transfer of the LTSP.

Output: Event message, at the P&S workstation, verifying the file transfer is invoked and a successful completion.

Expected Results: LTSP is resident within the P&S subsystem in the proper directory, file etc. with correct LTSP content.

Test Case 2 P&S Subsystem Ingest of Long Term Instrument Plan (LTIP)

This test verifies the P&S subsystem capability to ingest the Long Term Instrument Plan (LTIP) from the CSMS SMC. Upon receipt and store, the contents of the plan are verified.

Input: Coordination/Input at the Planner/Scheduler workstation to invoke the transfer of the LTIP.

Output: Event message, at the P&S workstation, verifying the file transfer is invoked and a successful completion.

Expected Results: LTIP is resident within the P&S subsystem in the proper directory, file etc. with correct LTIP contents.

Test Case 3 CSMS Transfer Attempt of Invalid LTSP to the FOS

This test verifies the P&S subsystem capability to flag/reject an LTSP due to invalid format. The subsystem will notify the P&S user via an event message that the LTSP format is invalid. The P&S subsystem will generate an event message to the CSMS SMC providing notification that the LTSP transfer has been rejected.

Input: Coordination/Input at the Planner/Scheduler workstation to invoke the transfer of an invalid LTSP. An LTSP file at the CSMS SMC with an invalid file format.

Output: An event message at the EOC Scheduler workstation notifying the user that the LTSP file has been rejected. An event message to the CSMS SMC notifying SMC that the file has been rejected.

Expected Results: Failed transfer will be written to an error log. Rejected LTSP can be viewed at the CSMS SMC to correct invalid format, parameters etc.

Test Case 4 CSMS Transfer Attempt of Invalid LTIP to the FOS

This test verifies the P&S subsystem capability to flag/reject an LTIP due to invalid format. The subsystem will notify the P&S user via an event message that the LTIP is invalid. The P&S subsystem will generate an event message to the CSMS SMC providing notification that the LTIP transfer has been rejected.

Input: Coordination/Input at the Planner/Scheduler workstation to invoke the transfer of an invalid LTIP. An LTSP file at the CSMS SMC that has an invalid file format.

Output: An event message at the Planner/Scheduler workstation notifying the user that the LTIP file has been rejected. An event message to the CSMS SMC notifying SMC that the file has been rejected.

Expected Results: Failed transfer will be written to an error log. Rejected LTIP can be viewed at the CSMS SMC to correct invalid format, parameters etc.

Test Case 5 EOC Ingest of Flight Dynamics Facility (FDF) Planning Data

This test verifies the P&S subsystem capability to ingest planning data from the Flight Dynamics Facility.

Input: Coordination/Input at the P&S workstation to transfer planning data from FDF to the EOC P&S Subsystem.

Output: Event message, at the P&S workstation, verifying the file transfer is invoked and a successful transfer completion.

Expected Results: FDF planning data files are resident as part of the P&S subsystem with correct orbit data content.

Test Case 6 FDF Transfer Attempt of Invalid Planning Data to the EOC

This test verifies the P&S subsystem capability to reject invalid planning data transferred from the FDF.

Input: Coordination/Input at the Planner/Scheduler workstation to invoke the transfer of FDF planning data. An invalid data file available for transfer attempt.

Output: Event message, at the P&S workstation, reporting the rejection of the data file. Event message to the FDF reporting file rejection.

Expected Results: Failed transfer attempt written to an error log. P&S acceptance of data file(s) only after corrections are made.

Test Case 7 P&S Subsystem Distribution of FDF Planning Data to CERES IST

This test verifies the P&S subsystem capability to distribute FDF planning data to the CERES PI/TL IST.

Input:	Coordination/Input at the Planner/Scheduler workstation to invoke transfer of the FDF files from the P&S Subsystem to the CERES PI/TL IST.
Output:	Event messages at the P&S workstation confirming the file transfer is invoked and a successful completion.
Expected Results:	FDF planning data resident at the CERES IST. CERES IST users are able to process the information and utilize same during instrument activity planning.

Test Case 8 P&S Subsystem Distribution of FDF Planning Data to MISR IST

This test verifies the P&S subsystem capability to distribute FDF planning data to the MISR PI/TL IST.

Input:	Coordination/Input at the Planner/Scheduler workstation to invoke transfer of the FDF files from the P&S Subsystem to the MISR PI/TL IST.
Output:	Event messages at the P&S workstation confirming the file transfer is invoked and a successful completion.
Expected Results:	FDF planning data resident at the MISR IST. MISR IST users are able to process the information and utilize same during instrument activity planning.

Test Case 9 P&S Subsystem Distribution of FDF Planning Data to MODIS IST

This test verifies the P&S subsystem capability to distribute FDF planning data to the MODIS PI/TL IST.

Input:	Coordination/Input at the Planner/Scheduler workstation to invoke transfer of the FDF files from the P&S Subsystemto the MODIS PI/TL IST.
Output:	Event messages at the P&S workstation confirming the file transfer is invoked and a successful completion.
Expected Results:	FDF planning data resident at the MODIS IST. MODIS IST users are able to process the information and utilize same during instrument activity planning.

Test Case 10 P&S Subsystem Distribution of FDF Planning Data to MOPITT IST

This test verifies the P&S subsystem capability to distribute FDF planning data to the MOPITT PI/TL IST.

Input:	Coordination/Input at the Planner/Scheduler workstation to invoke transfer of the FDF files from the P&S Subsystem to the MOPITT PI/TL IST.

Output: Event messages at the P&S workstation confirming the file transfer is invoked and a successful completion.

Expected Results: FDF planning data resident at the MOPITT IST. MOPITT IST users are able to process the information and utilize same during instrument activity planning.

Test Case 11 Distribution of FDF Planning Data to Command, Telemetry & Control (CT&C) Subsystem Engineering Workstation

This test verifies the P&S subsystem capability to distribute FDF planning data to the CT&C Subsystem Engineering Workstation.

Input: Coordination/Input at the P&S workstation to distribute FDF planning data files to the CT&C subsystem engineering workstation(s).

Output: Event messages in response to user inputs. Event message at the CT&C engineering workstation verifying the transfer of the data files to a CT&C workstation accessible directory.

Expected Results: FDF planning data accessible from the CT&C workstation(s).

Test Case 12 Distribution of FDF Planning Data to Guidance, Navigation & Control (GN&C) Subsystem Engineering Workstation

This test verifies the P&S subsystem capability to distribute FDF planning data to the GN&C Subsystem Engineering Workstation.

Input: Coordination/Input at the P&S workstation to distribute FDF planning data files to the GN&C subsystem engineering workstation(s).

Output: Event messages in response to user inputs. Event message at the CT&C engineering workstation verifying the transfer of the data files to a GN&C workstation accessible directory.

Expected Results: FDF planning data accessible from the GN&C workstation(s).

Test Case 13 Distribution of FDF Planning Data to Thermal Monitoring & Control (TM&C) Subsystem Engineering Workstation

This test verifies the P&S subsystem capability to distribute FDF planning data to the TM&C Subsystem Engineering Workstation.

Input: Coordination/Input at the P&S workstation to distribute FDF planning data files to the TM&C subsystem engineering workstation(s).

Output: Event messages in response to user inputs. Event message at the TM&C engineering workstation verifying the transfer of the data files to a TM&C workstation accessible directory.

Expected Results: FDF planning data accessible from the TM&C workstation(s).

Test Case 14 — Distribution of FDF Planning Data to Spacecraft Control Computer (SCC) Subsystem Engineering Workstation

This test verifies the P&S subsystem capability to distribute FDF planning data to the SCC Subsystem Engineering Workstation.

Input: Coordination/Input at the P&S workstation to distribute FDF planning data files to the SCC subsystem engineering workstation(s).

Output: Event messages in response to user inputs. Event message at the SCC engineering workstation verifying the transfer of the data files to a SCC workstation accessible directory.

Expected Results: FDF planning data accessible from the SCC workstation(s).

Test Case 15 — Distribution of FDF Planning Data to Power Monitoring & Control PM&C) Subsystem Engineering Workstation

This test verifies the P&S subsystem capability to distribute FDF planning data to the PM&C Subsystem Engineering Workstation.

Input: Coordination/Input at the P&S workstation to distribute FDF planning data files to the PM&C subsystem engineering workstation(s).

Output: Event messages in response to user inputs. Event message at the PM&C engineering workstation verifying the transfer of the data files to a PM&C workstation accessible directory.

Expected Results: FDF planning data accessible from the PM&C workstation(s).

Test Case 16 — EOC Generation of Mission Long Term Spacecraft Operations Plan

This test verifies the P&S capability to support generation of a Long-Term Spacecraft Operations Plan.

Input: Manual input at the Planner/Scheduler workstation. Valid LTSP and LTIP files available. Appropriate FDF orbital data files available.

Output: Event messages, at the Planner/Scheduler workstation verifying the initiation and successful completion of the spacecraft plan generation.

Expected Results: Valid spacecraft plan residing in a P&S subsystem directory. The spacecraft plan should be accessible by authorized users.

Test Case 17 — View, Modify and Deletion of Spacecraft Operations Plan by Authorized Users

This test verifies the P&S subsystem capability to support utility functions, in respect of the spacecraft operations plan, when invoked by an authorized user.

Input:	Input by an authorized user at the TBD workstation(s) to view the spacecraft operations plan, modify/edit the plan and delete a plan.
Output:	Event message at the authorized users workstation to verify a specific utility function has been invoked.
Expected Results:	The authorized user is able to view the spacecraft plan , verify any modifications to the plan and verify any deletion of the plan.

Test Case 18 "What If" Changes to Spacecraft Plan - Authorized Users

This test verifies the P&S subsystem capability to allow "what-if" changes to the spacecraft plan for the study of alternative schedules. The P&S subsystem shall allow for "what-if" changes without actually affecting the current spacecraft plan.

Input:	Input from an EOC or IST workstation TBD, to invoke the required "what-if" utility/capability.
Output:	Event messages, at the workstation to verify the utility has been invoked.
Expected Results:	Ability to make "what-if" changes to a "copy" of the spacecraft plan for study purposes. The actual spacecraft plan shall remain unaffected.

Test Case 19 Save, Delete and/or Retrieval of "What-if" Changes to Spacecraft Plan

This test verifies the P&S subsystem capability to support authorized users when saving a "what-if" session, deleting a "what-if" session and/or retrieval of a previously saved "what-if" session.

Input:	Input at a workstation TBD to save a previously generated "what-if" plan, delete a previously generated "what-if" plan and retrieve a previously saved "what-if" plan.
Output:	Event message at the workstation verifying the respective request has been invoked and successfully completed.
Expected Results:	The proper save of a "what-if" plan, the proper deletion of a "what-if" plan and the proper retrieval of a previously saved "what-if" plan.

5.1.24.2.2 Sequence 2 - Instrument Baseline Activity Profile (BAP) & BAP Deviations Lists

The following series of tests verifies the EOC capability to support generation of Instrument Baseline Activity Profiles for non-complex instruments. These test cases will verify the EOC capability to support BAP generation at the PI/TL IST and transfer of the BAP to the EOC Scheduler for planning & scheduling incorporation. This series of test cases also verify the capability to generate deviations to the BAPs, constraint checking of BAPs and that the EOC Scheduler is provided the capability to access BAPs for DAS incorporation.

Test Case 1 CERES IST Build of a CERES Baseline Activity Profile (BAP)

This test verifies that the EOC capability to support the generation of a CERES Baseline Activity Profile (BAP) utilizing the CERES IST(s).

Input: Input at the CERES IST workstation to invoke the BAP build/define utility. Input to create and save the CERES BAP.

Output: Event message at the CERES IST workstation verifying the build utility has been invoked. Event messages confirming the successful build and save of BAP.

Expected Results: CERES specific BAP saved in the appropriate directory, available for transfer to the EOC Scheduler for merging into DAS.

Test Case 2 MODIS IST Build of MODIS Baseline Activity Profile (BAP)

This test verifies the EOC capability to support the generation of a MODIS Baseline Activity Profile (BAP) at the MODIS PI/TL IST.

Input: Input at the MODIS IST workstation to invoke the BAP build/define utility. Input by the user to create the BAP and save the finished product.

Output: Event message, at the MODIS IST workstation verifying the build utility has been invoked. Event messages confirming the successful build and save of the MODIS BAP.

Expected Results: MODIS specific BAP saved in the appropriate directory, available for transfer to the EOC Scheduler for merging into the DAS.

Test Case 3 MOPITT IST Build of MOPITT Baseline Activity Profile (BAP)

This test verifies the EOC capability to support the generation of a MOPITT Baseline Activity Profile (BAP) at the MOPITT PI/TL IST.

Input: Input at the MOPITT IST workstation to invoke the BAP build/define utility. Input to create and save the MOPITT BAP.

Output: Event message at the MOPITT IST workstation verifying the build utility has been invoked. Event messages confirming the successful build and save of the BAP.

Expected Results: MOPITT specific BAP saved in the appropriate directory, available for transfer to the EOC Scheduler for merging into the DAS.

Test Case 4 MISR IST Build of MISR Baseline Activity Profile (BAP)

This test verifies the EOC capability to support the generation of a MISR Baseline Activity Profile (BAP) at the MISR PI/TL IST.

Input: Input at the MISR IST workstation to invoke the BAP build/define utility. Input to create and save the MISR BAP.

Output: Event message at the MISR IST workstation verifying the build utility has been invoked. Event messages confirming the successful build and save of the BAP.

Expected Results: MISR specific BAP saved in the appropriate directory, available for transfer to the EOC Scheduler for merging into the DAS.

Test Case 5 CERES PI/TL IST Transfer of BAP to the EOC Scheduler

This test verifies the EOC capability to support the transfer of a CERES BAP from the CERES IST to the EOC Scheduler for validation and incorporation into the DAS.

Input: Input/coordination at the CERES IST and/or EOC workstation to invoke the transfer of the CERES BAP to a directory that the EOC Scheduler is able to access.

Output: Event messages at the IST workstation and EOC Scheduler workstation in response to user inputs and status of transfer.

Expected Results: Successful transfer of CERES BAP file to a directory that the EOC Scheduler is able to access.

Test Case 6 MISR PI/TL IST Transfer of BAP to the EOC Scheduler

This test verifies the EOC capability to support the transfer of a MISR BAP from the MISR IST to the EOC Scheduler for validation and incorporation into the DAS.

Input: Input/coordination at the MISR IST and/or EOC workstation to invoke the transfer of the MISR BAP to a directory that the EOC Scheduler is able to access.

Output: Event messages at the IST workstation and EOC Scheduler workstation in response to user inputs and status of transfer.

Expected Results: Successful transfer of MISR BAP file to a directory that the EOC Scheduler is able to access.

Test Case 7 MODIS PI/TL IST Transfer of BAP to the EOC Scheduler

This test verifies the EOC capability to support the transfer of a MODIS BAP from the MODIS IST to the EOC Scheduler for validation and incorporation into the DAS.

Input: Input/coordination at the MODIS IST and/or EOC workstation to invoke the transfer of the MODIS BAP to a directory that the EOC Scheduler is able to access.

Output: Event messages at the IST workstation and EOC Scheduler workstation in response to user inputs and status of transfer.

Expected Results: Successful transfer of MODIS BAP file to a directory that the EOC Scheduler is able to access.

Test Case 8 MOPITT PI/TL IST Transfer of BAP to the EOC Scheduler

This test verifies the EOC capability to support the transfer of a MOPITT BAP from the MOPITT IST to the EOC Scheduler for validation and incorporation into the DAS.

Input: Input/coordination at the MOPITT IST and/or EOC workstation to invoke the transfer of the MOPITT BAP to a directory that the EOC Scheduler is able to access.

Output: Event messages at the IST workstation and EOC Scheduler workstation in response to user inputs and status of transfer.

Expected Results: Successful transfer of MOPITT BAP file to a directory that the EOC Scheduler is able to access.

Test Case 9 EOC Scheduler Access to BAPs Created by Instrument Planners

This test verifies that any BAP created by authorized user at an IST workstation can be accessed by the EOC Scheduler. Once EOC Scheduler access to the BAP(s) has been verified, testing proceeds to verify the EOC capability to modify the BAP(s) from the P&S workstation.

Input: Input/Coordination at the EOC P&S workstation to invoke the transfer of a BAP from IST(s) TBD. Input to modify the BAP and save the modified version without file management conflict.

Output: Event message, at the P&S workstation confirming the successful transfer of the BAP(s) to a directory that the Scheduler is able to access.

Expected Results: The EOC Scheduler shall successfully access the BAPs, modify the BAPs and save the modified version.

Test Case 10 EOC Capability to Constraint Check BAPs Against the Project Database

This test verifies the P&S subsystem capability to support constraint checking of a BAP by the instrument planner that created the BAP and/or the EOC Planner/Scheduler. This test verifies that all IST workstation have the capability to perform constraint checking.

Input: Input at the IST(s) workstation(s) TBD and/or the EOC P&S workstation to invoke the constraints database utility software. BAPs that are valid and BAPs that are known to have violations.

Output: Event messages at the IST and/or P&S workstation responding to user inputs/queries.

Expected Results: Those BAPs with known constraint violations will error during the checking cycle. The user shall be given an indication as to the violation and have a mechanism for correcting the violation.

Test Case 11 CERES IST Generation and Transfer of BAP Deviations

This test verifies the EOC capability to support the generation of CERES BAP Deviations at the CERES IST and transfer of the deviations lists to the EOC Scheduler.

Input: Input at the CERES IST workstation in invoke the required utility for generation of BAP deviations. Input/coordination at the CERES IST workstation to invoke the transfer of the deviations to the EOC Scheduler.

Output: Event messages at the CERES IST in response to user input. Event messages at the CERES IST in response the transfer of BAP deviations.

Expected Results: Successful generation of BAP deviations. Successful transfer of deviations to the EOC Scheduler for incorporation into the DAS.

Test Case 12 MISR IST Generation and Transfer of BAP Deviations

This test verifies the EOC capability to support the generation of MISR BAP Deviations at the MISR IST and transfer of the deviations lists to the EOC Scheduler.

Input: Input at the MISR IST workstation in invoke the required utility for generation of BAP deviations. Input/coordination at the MISR IST workstation to invoke the transfer of the deviations to the EOC Scheduler.

Output: Event messages at the MISR IST in response to user input. Event messages at the MISR IST in response the transfer of BAP deviations.

Expected Results: Successful generation of BAP deviations. Successful transfer of deviations to the EOC Scheduler for incorporation into the DAS.

Test Case 13 MODIS IST Generation and Transfer of BAP Deviations

This test verifies the EOC capability to support the generation of MODIS BAP Deviations at the MODIS IST and transfer of the deviations lists to the EOC Scheduler.

Input: Input at the MODIS IST workstation in invoke the required utility for generation of BAP deviations. Input/coordination at the MODIS IST workstation to invoke the transfer of the deviations to the EOC Scheduler.

Output: Event messages at the MODIS IST in response to user input. Event messages at the MODIS IST in response the transfer of BAP deviations.

Expected Results: Successful generation of BAP deviations. Successful transfer of deviations to the EOC Scheduler for incorporation into the DAS.

Test Case 14 MOPITT IST Generation and Transfer of BAP Deviations

This test verifies the EOC capability to support the generation of MOPITT BAP Deviations at the MOPITT IST and transfer of the deviations lists to the EOC Scheduler.

Input: Input at the MOPITT IST workstation in invoke the required utility for generation of BAP deviations. Input/coordination at the MOPITT IST workstation to invoke the transfer of the deviations to the EOC Scheduler.

Output: Event messages at the MOPITT IST in response to user input. Event messages at the MOPITT IST in response the transfer of BAP deviations.

Expected Results: Successful generation of BAP deviations. Successful transfer of deviations to the EOC Scheduler for incorporation into the DAS.

Test Case 15 EOC Scheduler Access and Constraint Check of BAP Deviations

This test verifies the EOC capability to support the EOC Scheduler access to the BAP deviations generated by instrument PI/TLs at respective ISTs. Once the capability of the EOC Scheduler to access the deviations has been verified, this test case verifies the capability of the EOC to support EOC Scheduler constraint checking of the deviations against the project database.

Input: Input at the EOC Scheduler workstation to access BAP deviations TBD. Input at the EOC workstation to invoke the required utility for constraint checking of the deviations against the project database.

Output: Event messages in response to user inputs. Report generation in response to constraint check of BAP deviations.

Expected Results: Successful access to BAP deviations from the EOC Scheduler workstation. Successful constraint check of BAP deviations against the project database.

Test Case 16 EOC Scheduler Generation of Instrument Resource Profile

This test verifies the EOC capability to support the generation of Instrument Resource Profiles based on Instrument BAPs and BAP deviations submitted to the EOC Scheduler.

Input: Instrument BAPs and BAP resource deviations TBD. Input at the EOC Scheduler workstation to invoke the utility required for Instrument Resource Profile generation.

Output: Event messages in response to user inputs. Report of IRP generation results.

Expected Results: Successful generation of an Instrument Resource Profile, available for use by authorized users.

Test Case 17 EOC Scheduler Generation of Instrument Activity Profile

This test verifies the EOC capability to support the generation of Instrument Activity Profiles based on Instrument BAPs and BAP deviations submitted to the EOC Scheduler.

Input: Instrument BAPs and BAP activity deviations TBD. Input at the EOC Scheduler workstation to invoke the utility required for Instrument Activity Profile generation.

Output: Event messages in response to user inputs. Report of IAP generation results.

Expected Results: Successful generation of an Instrument Activity Profile, available for use by authorized users.

5.1.24.2.3 Sequence 3 - Spacecraft Subsystem Resource Profiles & Resource Deviations

The following series of tests verify the EOC capability to support the generation of Spacecraft Subsystem Resource Profile(s) and Resource Deviation Lists. Testing will verify the capability to generate subsystem specific profiles that the EOC Scheduler is able to access for incorporation into final scheduling. This sequence of test cases will also verify the capability to generate deviations to the profiles for EOC Scheduler incorporation into final scheduling.

Test Case 1 EOC Generation of SSRP to include Command, Telemetry & Control (CT&C) Resource Needs

This test verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes CT&C specific resource needs.

Input: CT&C specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include CT&C specific resource information. SSRP available to authorized users.

Test Case 2 EOC Generation of SSRP to include Guidance, Navigation & Control (GN&C) Resource Needs

This test verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes GN&C specific resource needs.

Input: GN&C specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include GN&C specific resource information. SSRP available to authorized users.

Test Case 3 EOC Generation of SSRP to include Thermal Control Subsystem (TCS) Resource Needs

This test verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes TCS specific resource needs.

Input: TCS specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to user inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include TCS specific resource information. SSRP available to authorized users.

Test Case 4 EOC Generation of SSRP to include Electrical Power Subsystem (EPS) Resource Needs

This test verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes EPS specific resource needs.

Input: EPS specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to user inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include EPS specific resource information. SSRP available to authorized users.

Test Case 5 EOC Generation of SSRP to include Spacecraft Control Computer (SCC) Resource Needs

This test case verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes SCC specific resource needs.

Input: SCC specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to user inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include SCC specific resource information. SSRP available to authorized users.

Test Case 6 — EOC Generation of SSRP to include Solid State Recorder (SSR) Resource Needs

This test verifies the EOC capability to support the generation of a Spacecraft Subsystem Resource Profile that includes SSR specific resource needs.

Input: SSR specific resource scheduling information. Input at the EOC workstation TBD to invoke the utility required for SSRP generation.

Output: Event messages in response to inputs. Report of SSRP generation results.

Expected Results: Successful generation of SSRP to include SSR specific resource information. SSRP available to authorized users.

Test Case 7 — EOC Generation of Spacecraft Subsystem Resource Deviation Lists

This test verifies the EOC capability to support generation of Subsystem Resource Deviations based on deviations for a given subsystem (i.e., CT&C, GN&C, EPS, TCS, SCC and SRR).

Input: SSRP available at the EOC. Deviations to the SSRP for subsystems TBD. Input at the EOC workstation TBD to invoke the necessary utility for generation of subsystem resource deviations.

Output: Event messages in response to user requests. Report of resource deviation generation results.

Expected Results: Successful generation of subsystem deviations. Successful replacement of original SSRP with updates due to deviations. Updates available to authorized users.

5.1.24.2.4 Sequence 4 - EOC FOS - ASTER ICC/IST Connectivity

The following sequence of tests verify the capability of the EOC to support the EOC - ASTER ICC interface. Due to the limited functionality of this interface in TRMM (Rel A), testing will verify connectivity only with the passing of test messages between the two facilities.

Test Case 1 — EOC FOS Transfer of Test Message to the ASTER ICC

This test verifies the EOC capability to support the generation of a "handshake" test message and transfer of the test message to the ASTER ICC.

Input: Input/coordination at the EOC workstation TBD to enable the interface to the ASTER ICC. Input at the EOC workstation to invoke the generation and transfer of an ASTER test message.

Output: Test message available for transfer to ASTER ICC. Event messages in response to user input, generation and transfer of test

message. Message transmit counter to increment on display page TBD.

Expected Results: Successful transfer of the test message to the ASTER ICC. Receipt of an ASTER acknowledgment message at the EOC.

Test Case 2 EOC Receipt of Test Message Generated at the ASTER ICC

This test verifies the EOC capability to accept test messages generated and transferred from the ASTER ICC.

Input: Input/coordination at the ASTER ICC to generate and transfer a test message. Input at the EOC workstation TBD to enable the FOS for receipt of the ASTER test message.

Output: Event messages in response to user input. Event message reporting receipt of ASTER test message. EOC acknowledgment message to ASTER ICC.

Expected Results: Successful receipt of ASTER test message at the EOC. Increment of message received counter on EOC display page TBD.

5.1.24.2.5 Sequence 5 - EOC FOS - GSFC Network Control Center (NCC) Connectivity

The following series of tests verify the connectivity of the EOC to NCC interface. Specifically, testing will verify that the EOC FOS is capable of generating and sending a "handshake" test message to the NCC. Upon verification of EOC - NCC connectivity, this sequence of test cases will verify the capability of the EOC FOS to generate, send and negotiate TDRSS Contact Requests Schedules to the NCC for approval and/or denial.

Test Case 1 EOC FOS Transfer of Test Message to the NCC

This test verifies the Planning & Scheduling subsystem supports the capability to send a test message to the NCC via ECOM.

Input: Input/Coordination at the P&S workstation to send a test message to the NCC.

Output: Event message, at the P&S workstation, confirming the test message was sent.

Expected Results: Successful receipt of the test message by the NCC. Successful receipt of an NCC acknowledgment message by the EOC/P&S.

Test Case 2 EOC FOS Receipt of Test Message Sent by the NCC

This test verifies P&S subsystem capability to accept a test message sent by the NCC via ECOM.

Input: Coordination between the P&S test conductor and NCC to send a test message from the NCC to the EOC/P&S via ECOM.

Output: Event message at the P&S workstation reporting receipt of the test message. On the P&S display page TBD, the counter for NCC messages received will increment.

Expected Results: Successful receipt of the test message by the EOC/P&S. Capability at the EOC/P&S to return an acknowledgment message to the NCC.

Test Case 3 EOC Generation of TDRSS Contact Requests Schedule

This tests verifies the P&S subsystem capability to support the generation of a TDRSS Contact Request Schedule. Based on spacecraft subsystem and instrument resources, activity etc., the P&S will generate a schedule of time periods in which contact with the AM-1 spacecraft (via TDRSS) is desired.

Input: Input/Coordination at the P&S subsystem workstation to generate a valid TDRSS contact requests schedule.

Output: Appropriate event messages responding to schedule generation session. Event message verifying schedule has been saved and is available for transfer to the NCC.

Expected Results: Successful generation of the TDRSS contact schedule. Ability to edit, save and delete the schedule file. Ability to transfer the file to the NCC.

Test Case 4 Authorized User Access to the TDRSS Contact Request Schedule

This test verifies that the P&S subsystem supports access to the TDRSS contact requests schedule by those users with appropriate authorization. This test in turn verifies that those users without proper authorization cannot access the schedule.

Input: Input at the EOC workstation(s) TBD to display, edit, save, delete or printout a copy of the TDRSS requests schedule.

Output: Event messages confirming the requested function for authorized users. Event message denying requests/access to unauthorized users.

Expected Results: Those users with authorization will successfully display, edit, save, delete and print the schedule. Requests from unauthorized users shall be rejected.

Test Case 5 EOC Transfer of the TDRSS Contact Requests Schedule to the NCC

This test verifies the P&S subsystem capability to support the transfer of the TDRSS Contact Requests Schedule to the NCC for approval and/or disapproval.

Input: Input/Coordination to transfer the requests schedule to the NCC.

Output: Event message verifying the system initiation and completion of the file transfer.

Expected Results: A successful transfer of the schedule. An acknowledgment message from the NCC confirming receipt of the schedule.

Test Case 6 EOC FOS Ingest of NCC Accepted TDRSS Contact Requests

This test verifies the P&S subsystem capability to ingest, from the NCC, accepted/approved TDRSS contact requests.

Input: Input/Coordination at the Planner/Scheduler workstation to ingest the approved schedule file transferred from the NCC.

Output: Event message, at the P&S workstation confirming receipt of the schedule and successful completion of the ingest.

Expected Results: A valid TDRSS contact schedule in the proper P&S directory. The schedule shall be available/accessible for any authorized users.

Test Case 7 EOC FOS Ingest of NCC Rejected TDRSS Contact Requests

This test verifies the P&S subsystem capability to ingest, from the NCC, those contact requests that have been rejected and require rescheduling.

Input: Input/Coordination at the Planner/Scheduler workstation to ingest the schedule of rejected contact requests from the NCC.

Output: Event message, at the P&S workstation confirming receipt of the schedule and successful completion of the ingest.

Expected Results: The schedule of rejected contacts shall reside in a P&S directory without interfering with the approved schedule. The rejected contact requests shall be available/accessible for any authorized users.

Test Case 8 EOC FOS Reschedule of NCC Rejected TDRSS Contact Requests

This test verifies the P&S subsystem capability to support the rescheduling of NCC rejected TDRSS contact requests.

Input: Input/Coordination at the Planner/Scheduler workstation to generate a TDRSS contact requests schedule to include the rescheduling of previously rejected contact requests.

Output: Event message, at the P&S workstation, verifying the generation of the TDRSS contact request schedule. Event message confirming successful completion of the generation.

Expected Results: A TDRSS Contact Requests Schedule residing in the proper P&S directory and available for transfer to the NCC.

5.1.24.2.6 Sequence 6 - Preliminary Resource Schedule (PRS)

The following series of tests verify the EOC capability to support the generation of a Preliminary Resource Schedule. Testing will verify that the EOC Scheduler is provided with the required FOS tools for generation of a Preliminary Resource Schedule based on an available

Instrument Resource Profile, Spacecraft Subsystem Resource Profile and an approved TDRSS Contact Schedule.

Test Case 1 EOC Generation of a Preliminary Resource Schedule (PRS)

This test verifies the EOC capability to support the generation of a PRS. Testing verifies that the capability exist that allows the EOC scheduler to build a preliminary resource schedule based on an IRP, SSRP and TDRSS Contact Scheduling as input.

Input: IRP, SSRP, TDRSS Schedule. Input at the EOC scheduler workstation to invoke the required utility for PRS generation.

Output: Event messages in response to user inputs. PRS generation report.

Expected Results: Successful generation of a valid Preliminary Resource Schedule.

Test Case 2 Authorized User Access to the Preliminary Resource Schedule

This test case verifies the EOC capability to support authorized EOC and IST user access to the Preliminary Resource Schedule.

Input: Valid Preliminary Resource Schedule. Input at EOC and IST workstations TBD to verify the capability to access the contents of the PRS.

Output: Event messages in response to user input.

Expected Results: Successful access to the directory/file containing the PRS from EOC and IST workstations TBD.

Test Case 3 EOC Generation of Spacecraft Subsystem Activity List

This test case verifies the EOC capability to support the generation of a Spacecraft Subsystem Activity List.

Input: Existing detailed activity schedule, preliminary resources schedule, identified spacecraft subsystem activities. Input at the EOC Scheduler workstation to invoke the necessary tools/utility.

Output: Event messages in response to user input. Hardcopy of Spacecraft Subsystem Activity List.

Expected Results: Successful generation of the Spacecraft Subsystem Activity List with accurate information relative to data input.

5.1.24.2.7 Sequence 7 - Detailed Activity Schedule (DAS)

The following series of tests verify the EOC capability to generate a valid, conflict free Detailed Activity Schedule (DAS), that details subsystem and instrument activity for a specified target day. Once the DAS has been generated, testing verifies the EOC capability to perform an activity constraint check against the project database, access to the DAS by authorized users and the capability to update the DAS based on Late Changes and Targets of Opportunity (TOOs).

Test Case 1 EOC Generation of a Detailed Activity Schedule

This test verifies the EOC capability to support the generation of a Detailed Activity Schedule based on information in the Preliminary Resource Schedule, Instrument Deviation Lists and Subsystem Deviation Lists.

Input: Preliminary Resource Schedule. Input at the EOC Scheduler workstation to invoke the required utility for DAS generation.

Output: Event messages in response to user inputs. DAS generation report.

Expected Results: Successful generation of a Detailed Activity Schedule.

Test Case 2 EOC Activity Constraint Check of the Detailed Activity Schedule

This test verifies the EOC capability to perform an activity constraint check of a generated Detailed Activity Schedule.

Input: Valid conflict-free DAS. DAS with known activity conflicts, constraint violations. Input at the EOC Scheduler workstation to invoke the required constraint check utility.

Output: Event messages in response to user inputs. Event messages in response to constraint check against the project database. Report of constraint check results.

Expected Results: Successful check of conflict free DAS. Error/alarm messages and appropriate reporting of constraint violations for DAS with known conflicts.

Test Case 3 Authorized EOC and IST User Access to Detailed Activity Schedule

This test verifies the EOC capability to support authorized user access to the DAS. Authorized users at EOC and IST workstations TBD will attempt to access the proper directory/file in which the DAS resides.

Input: Valid DAS available in appropriate directory/file. Input at the EOC and IST user workstations TBD in an effort to access the DAS.

Output: Event messages in response to user input. Display of DAS contents at the workstation monitor.

Expected Results: Successful access to the DAS contents by authorized users. Appropriate rejection of access attempts by unauthorized users.

Test Case 4 EOC Updates to an Existing Detailed Activity Schedule

This test verifies the EOC capability to generate updates to a DAS based on Late Changes and/or Targets Of Opportunity (TOOs).

Input:	Late Changes or TOOs TBD. Valid DAS available for updating. Input at the EOC Scheduler workstation to invoke the required utility for updating of the DAS.
Output:	Event messages in response to user input. Report detailing results of DAS update generation.
Expected Results:	Successful update of DAS to reflect Late Changes and/or TOOs.

5.1.25 FOS Command Management Thread (T14-05)

5.1.25.1 Thread Objectives

The objective of the command management thread is to identify the FOS Command Management Subsystem (CMS) functionality required for the generation of spacecraft and instrument command loads based on the contents of a valid Detailed Activity Schedule (DAS). This thread identifies the functionality required for an interpretation of the planning & scheduling information available in the DAS and preparation of the DAS information into the corresponding command loads and ground script. Specifically, this thread identifies the following Command Management functionality as it relates to the FOS TRMM (Rel A) Build:

- Command-level Constraint Checking of the Detailed Activity Schedule (DAS).
- Build of Absolute Time Commands (ATCs) from the contents of the DAS.
- Build of Flight Operations Ground Script from the contents of the DAS.
- Generation of ATC Uplink Load and Determination of Uplink Window.
- Updates due to DAS Late Changes.
- EOC Ingest of Instrument Microprocessor Load Contents.
- EOC Build/Ingest of Relative Time Command (RTC) Load Contents.
- EOC Build/Ingest of Table Load Contents.

5.1.25.2 Thread Test Description

For a valid, conflict-free Detailed Activity Schedule (DAS), the Command Management Subsystem (CMS) functionality includes the generation of Absolute Time Commands and the Flight Operations Ground Script that correspond to the contents of the DAS.

Command Management Thread testing is initiated with the verification of the CMS functionality required for command-level constraint checking of the contents of the DAS. For a valid conflict-free DAS generated during the Planning & Scheduling subsystem testing, the Command Management Subsystem utility/object is invoked and the required check of the DAS contents against the command definitions of the project database is performed. Testing will verify the capability to successfully constraint check the DAS and perform the necessary corrections should violations be detected.

Following the completion of command-level constraint check testing, thread testing continues to verify the CMS capability to support the generation of Absolute Time Command (ATC) contents as well as the generation of the corresponding Ground Script. Testing is performed from a flight

operations user view point and verifies that the functionality allows the user to successfully produce a valid ATC and ground script.

Following the completion of ATC and ground script testing, thread testing continues to verify that CMS TRMM (Rel A) functionality supports requirements related to the incorporation of Late Changes and Targets of Opportunity (TOOs) into the Detailed Activity Schedule (DAS). The effect of the late changes on the DAS will be reflected in ATCs that have not been uplinked to the AM-1 spacecraft. Thread testing will verify that from a flight operations users view point, the CMS provides the capability to update the ATCs and Ground Scripts effected by the changes.

The final phase of Command Management Thread testing verifies the CMS TRMM (Rel A) functionality related to the EOC ingest of instrument microprocessor load content, Relative Time Command (RTC) load content and Table Load content for validation and uplink load generation. Thread testing in this phase will verify from a flight operations user view point that these three load types can be successfully transferred to the EOC, compared against the project database for load content ID validation and then processed into the required uplink load format.

Dependencies: (If Applicable)

In order to verify the objectives of the command management thread, the following interface and functional capabilities are required:

- Planning & Scheduling Thread
- EOC - PI/TL IST Interface
- EOC - ASTER ICC/IST Interface
- Spacecraft Commanding Thread

Test Support Requirements:

- Hardware:
  - TBS
- Software:
  - TBS
- Data:
  - TBS

5.1.25.2.1 Sequence 1 - CMS Command-level Constraint Check of the DAS

The following series of tests verify requirements related to the EOC capability to support command-level constraint checking of the Detailed Activity Schedule (DAS) against the command definitions found in the project database.

Test Case 1 — Detailed Activity Schedule (DAS) Command-level Constraint Check

This test verifies requirements related to the EOC capability to compare the commanding activity in the DAS to command definitions in the project database.

Input: Valid DAS generated during Planning & Scheduling subsystem testing. A DAS with known command constraint violations. Input at the EOC Scheduler workstation to invoke the required constraint check utility.

Output: Event messages in response to user input. Report of constraint check results

Expected Results: Successful check of DAS command activity. Error/alarm flag for DAS with known command constraint violations.

5.1.25.2.2 Sequence 2 - Absolute Time Command (ATC) Load and Ground Script Generation

The following series of tests verify requirements related to EOC capability to support the generation of an ATC load and Flight Operations Ground Script based on the contents of a Detailed Activity Schedule (DAS).

Test Case 1 Command Management Build of Absolute Time Commands (ATCs)

This test verifies requirements related to the EOC capability to build the Absolute Time Commands (ATCs) that correspond to the instrument and spacecraft subsystem activity included in the Detailed Activity Schedule (DAS).

Input: Various instrument and spacecraft subsystem command activities included in a valid DAS. Input at the EOC workstation TBD to invoke ATC build.

Output: Event messages in response to user input. ATC file available for output to a printer.

Expected Results: Successful build of Absolute Time Commands (ATCs) that include various instrument and subsystem commands.

Test Case 2 CMS Generation of Uplink Load for Absolute Time Commands

This test verifies requirements related to the EOC capability to generate an uplink load for absolute time commands.

Input: Valid Absolute Time Commands built from the contents of a valid DAS. Input at the EOC workstation TBD to invoke the required utility.

Output: Event messages in response to user inputs. ATC uplink load file.

Expected Results: Absolute Time Commands formatted into the uplink load and available to the Command Subsystem for uplink to the spacecraft.

Test Case 3 Determine an ATC Uplink Load into an Uplink Window

This test verifies requirements related to the EOC capability to support the allocation of an ATC load to an acceptable uplink window.

Input:	Valid ATC uplink load. Input at the EOC workstation TBD to invoke required utility.
Output:	Event messages in response to user input. Report of ATC uplink window assigned.
Expected Results:	Successful assignment of ATC uplink load to an acceptable uplink window that is reflected in the operational ground script.

Test Case 4 CMS Generation of Flight Operations Ground Script

This test verifies requirements related to the EOC capability to support the generation of an operational ground script detailing the time ordered directives to be performed for spacecraft and instrument commanding as well as ground system configurations.

Input:	DAS and associated ATC load. Input at the EOC workstation TBD to invoke the required utility.
Output:	Event messages in response to user inputs. Ground Script file available for view, print etc. by authorized users.
Expected Results:	Successful generation of EOC ground script detailing the time ordered directives to take place during uplink of the ATC load.

Test Case 5 EOC User Interface Display of Ground Script - Authorized Users

This test verifies requirements related to the EOC capability to support the display of the ground script by authorized users.

Input:	Valid ground script available in the appropriate directory. Input at EOC/IST workstations TBD to display the contents of the ground script at an authorized users workstation.
Output:	Event messages in response to user inputs.
Expected Results:	Successful display of the ground script on the workstation monitors of authorized users.

Test Case 6 EOC Generation and Storage of Spacecraft Commands for Emergency Situations

This test verifies requirements related to the EOC capability to support generation and storage of specific commands utilized during spacecraft emergency situations.

Input:	Valid Detailed Activity Schedule (DAS) with emergency command specific activities included. Input at the EOC Scheduler workstation to invoke tools necessary for command generation.
Output:	Event messages at the Scheduler workstation in response to user input.
Expected Results:	Successful generation of emergency commands. Capability to store commands and access for uplink on demand.

5.1.25.2.3 Sequence 3 - CMS Support of Detailed Activity Schedule Late Changes and TOOs

The following series of tests verify requirements related to the EOC capability to support updates to the Detailed Activity Schedule as a result of instrument and/or spacecraft late changes or Targets of Opportunity. This series will verify the capability of the EOC to accept the late changes/TOOs, constraint check the updates against the project database and incorporate the changes into the most recent version of the DAS.

Test Case 1 — Command-level Constraint Check of Late Changes to the DAS

This test verifies requirements related to the EOC capability to support late changes to the Detailed Activity Schedule (DAS).

Input: Valid DAS available. Various instrument and/or subsystem deviations TBD to incorporate into the DAS as late changes. Input at the EOC workstation TBD to perform the required command constraint check against command definitions in the project database.

Output: Event messages in response to user inputs. Report of command constraint check results.

Expected Results: Successful constraint check of late changes. Error/alarm messages or report if known violations are included in test case.

Test Case 2 — Late Change Updates to an Absolute Time Command Load

This test verifies requirements related to the EOC capability to support late change updates to an ATC load that has yet to be uplinked to the spacecraft.

Input: Valid ATC load available for uplink. Input at the EOC workstation TBD to invoke the required utility for ATC update.

Output: ATC load file.

Expected Results: Successful generation of ATC load file that reflects late change updates. Capability to manage updated ATC versus older version.

Test Case 3 — Generation of ATC Uplink Load for Late Change Updates

This test verifies requirements related to the EOC capability to support generation of an uplink load for an ATC that has been updated due to late changes.

Input: Valid ATCs that have been updated due to late changes. Input at the EOC workstation TBD to invoke the required utility for ATC uplink load generation.

Output: Event messages in response to user input. ATC uplink load file generated.

Expected Results: Successful generation of an updated ATC uplink load. Capability to manage the uplink load versus the older version.

Test Case 4 Modification of Ground Script to Reflect Late Change Updates

This test verifies requirements related to the EOC capability to support modification of the flight operations ground script to reflect instrument and/or subsystem commanding late changes.

Input: Valid flight operations ground script available. Input at the EOC workstation TBD to invoke the required utility for ground script modification.

Output: Event messages in response to user input. Ground script file reflecting modifications due to late changes.

Expected Results: Successful modification of ground script. Capability to manage to modified version of ground script versus older versions.

5.1.25.2.4 Sequence 4 - EOC Load Content Ingest and Load Management

The following series of tests verify requirements related to the EOC capability to ingest and manage instrument microprocessor load content, RTC load content and Table load contents. Once ingested, this sequence of test cases verifies the EOC capability to validate the Load content ID and generate the necessary uplink load for uplink to the AM-1 spacecraft via EDOS.

Test Case 1 EOC Ingest of Instrument Microprocessor Load from CERES SCF

This test verifies requirements related to the EOC capability to ingest instrument microprocessor loads transferred from the CERES Science Computing Facility (SCF).

Input: Valid CERES microprocessor load TBD. Input/coordination at the EOC workstation TBD to transfer the CERES microprocessor load from the SCF to the EOC.

Output: Event messages at the user workstation in response to user input.

Expected Results: Successful transfer of the CERES instrument microprocessor load to the appropriate EOC directory.

Test Case 2 EOC Ingest of Instrument Microprocessor Load from MISR SCF

This test verifies requirements related to the EOC capability to ingest instrument microprocessor loads transferred from the MISR Science Computing Facility (SCF).

Input: Valid MISR microprocessor load TBD. Input/coordination at the EOC workstation TBD to transfer the MISR microprocessor load from the SCF to the EOC.

Output: Event messages at the user workstation in response to user input.

Expected Results: Successful transfer of the MISR instrument microprocessor load to the appropriate EOC directory.

Test Case 3 EOC Ingest of Instrument Microprocessor Load from MODIS SCF

This test verifies requirements related to the EOC capability to ingest instrument microprocessor loads transferred from the MODIS Science Computing Facility (SCF).

Input: Valid MODIS microprocessor load TBD. Input/coordination at the EOC workstation TBD to transfer the MODIS microprocessor load from the SCF to the EOC.

Output: Event messages at the user workstation in response to user input.

Expected Results: Successful transfer of the MODIS instrument microprocessor load to the appropriate EOC directory.

Test Case 4 EOC Ingest of Instrument Microprocessor Load from MOPITT SCF

This test verifies requirements related to the EOC capability to ingest instrument microprocessor loads transferred from the MOPITT Science Computing Facility (SCF).

Input: Valid MOPITT microprocessor load TBD. Input/coordination at the EOC workstation TBD to transfer the MOPITT microprocessor load from the SCF to the EOC.

Output: Event messages at the user workstation in response to user input.

Expected Results: Successful transfer of the MOPITT instrument microprocessor load to the appropriate EOC directory.

Test Case 5 EOC Validation of CERES Instrument Microprocessor Content ID

This test verifies requirements related to the EOC capability to verify the content ID of a CERES instrument microprocessor load after transfer from the SCF to the EOC.

Input: CERES instrument microprocessor load TBD. CERES instrument microprocessor load with a known invalid content ID. Input at the EOC workstation TBD to invoke the required utility for validation of the loads content ID.

Output: Event messages in response to user inputs. Report/file containing content ID validation results.

Expected Results: Successful validation of content ID. Error/alarm message for invalid content ID.

Test Case 6 EOC Validation of MISR Instrument Microprocessor Content ID

This test verifies requirements related to the EOC capability to verify the content ID of a MISR instrument microprocessor load after transfer from the SCF to the EOC.

Input: MISR instrument microprocessor load TBD. MISR instrument microprocessor load with a known invalid content ID. Input at the

EOC workstation TBD to invoke the required utility for validation of the loads content ID.

Output: Event messages in response to user inputs. Report/file containing content ID validation results.

Expected Results: Successful validation of content ID. Error/alarm message for invalid content ID.

Test Case 7 EOC Validation of MODIS Instrument Microprocessor Content ID

This test verifies requirements related to the EOC capability to verify the content ID of a MODIS instrument microprocessor load after transfer from the SCF to the EOC.

Input: MODIS instrument microprocessor load TBD. MODIS instrument microprocessor load with a known invalid content ID. Input at the EOC workstation TBD to invoke the required utility for validation of the loads content ID.

Output: Event messages in response to user inputs. Report/file containing content ID validation results.

Expected Results: Successful validation of content ID. Error/alarm message for invalid content ID.

Test Case 8 EOC Validation of MOPITT Instrument Microprocessor Content ID

This test verifies requirements related to the EOC capability to verify the content ID of a MOPITT instrument microprocessor load after transfer from the SCF to the EOC.

Input: MOPITT instrument microprocessor load TBD. MOPITT instrument microprocessor load with a known invalid content ID. Input at the EOC workstation TBD to invoke the required utility for validation of the loads content ID.

Output: Event messages in response to user inputs. Report/file containing content ID validation results.

Expected Results: Successful validation of content ID. Error/alarm message for invalid content ID.

Test Case 9 EOC Generation of CERES Instrument Microprocessor Uplink Load

This test verifies requirements related to the EOC capability to support the proper generation of an uplink load for a CERES instrument microprocessor load transferred from the CERES SCF.

Input: CERES instrument microprocessor load TBD. Input at the EOC workstation TBD to invoke the required utility to generate the necessary uplink load format.

Output: CERES microprocessor uplink load file accessible to authorized users. Report of load generation statistics/results.

Expected Results: Successful generation of an uplink load to be verified through analysis.

Test Case 10 EOC Generation of MISR Instrument Microprocessor Uplink Load

This test case verifies requirements related to the EOC capability to support the proper generation of an uplink load for a MISR instrument microprocessor load transferred from the MISR SCF.

Input: MISR instrument microprocessor load TBD. Input at the EOC workstation TBD to invoke the required utility to generate the necessary uplink load format.

Output: MISR microprocessor uplink load file accessible to authorized users. Report of load generation statistics/results.

Expected Results: Successful generation of an uplink load to be verified through analysis.

Test Case 11 EOC Generation of MODIS Instrument Microprocessor Uplink Load

This test verifies requirements related to the EOC capability to support the proper generation of an uplink load for a MODIS instrument microprocessor load transferred from the MODIS SCF.

Input: MODIS instrument microprocessor load TBD. Input at the EOC workstation TBD to invoke the required utility to generate the necessary uplink load format.

Output: MODIS microprocessor uplink load file accessible to authorized users. Report of load generation statistics/results.

Expected Results: Successful generation of an uplink load to be verified through analysis.

Test Case 12 EOC Generation of MOPITT Instrument Microprocessor Uplink Load

This test verifies requirements related to the EOC capability to support the proper generation of an uplink load for a MOPITT instrument microprocessor load transferred from the MOPITT SCF.

Input: MOPITT instrument microprocessor load TBD. Input at the EOC workstation TBD to invoke the required utility to generate the necessary uplink load format.

Output: MOPITT microprocessor uplink load file accessible to authorized users. Report of load generation statistics/results.

Expected Results: Successful generation of an uplink load to be verified through analysis.

Test Case 13 Build of Relative Time Command Load Contents By Authorized Users

This test verifies requirements related to the EOC capability to support the creation of Relative Time Command load contents by authorized users located at external IST locations and within the EOC.

Input: Command Activity TBD. Input at the external IST(s) and/or internal user stations TBD, to invoke the RTS Load Builder utility.

Output: Event messages in response to user input. RTC load contents to include appropriate time tags and load mnemonics.

Expected Results: Successful build of RTC load contents TBD. Load content(s) available for transfer to the EOC.

Test Case 14 EOC Ingest of Relative Time Command (RTC) Load Content from External IST(s)

This test case verifies requirements related to the EOC capability to support the receipt of Relative Time Commands (RTC) created at external IST(s) location(s) TBD.

Input: Relative Time Command load contents generated at external IST locations TBD. Input/coordination at the EOC workstation TBD to invoke the transfer of the RTC from the IST to the appropriate EOC directory.

Output: Event messages in response to user inputs. Event messages in response to transfer attempt.

Expected Results: Successful transfer of the RTC load file to the appropriate EOC directory.

Test Case 15 Transfer of Relative Time Command (RTC) Load Content from EOC Authorized Users

This test verifies requirements related to the EOC capability to support the transfer of Relative Time Commands (RTC) created by authorized users in the EOC TBD, to the Command Management Subsystem for content ID validation and uplink load generation.

Input: Relative Time Command load contents generated by EOC authorized user TBD. Input/coordination at the EOC workstation TBD to invoke the transfer of the RTC load contents to an appropriate EOC directory.

Output: Event messages in response to user inputs. Event messages in response to transfer attempt.

Expected Results: Successful transfer of the RTC load content to the EOC. Successful validation of load content ID and uplink load generation by the CMS.

Test Case 16 EOC Validation of the Relative Time Command Content ID

This test verifies requirements related to the EOC capability to support the validation of the content ID for an RTC load received at the EOC.

Input: Recently ingested RTC load available in the appropriate EOC directory. Input at the EOC workstation TBD to invoke the utility required to validate the content ID.

Output: Event messages in response to user input. Report of content ID validation results.

Expected Results: Successful validation of RTC load content ID. Error/alarm messages/report for known invalid content ID.

Test Case 17 EOC Uplink Load Generation for Relative Time Commands

This test case verifies requirements related to the EOC capability to generate an uplink load format for RTCs for uplink to the spacecraft.

Input: Recently ingested RTC load with a valid load content ID. Input at the EOC workstation TBD to invoke the generation of the uplink load.

Output: Event messages in response to user inputs. Report of load generation statistics/results.

Expected Results: Successful generation of RTC uplink load file.

Test Case 18 Build of Table Load Contents by Authorized Users

This test verifies requirements related to the EOC capability to support the creation of Table load contents by authorized users located at external IST locations and within the EOC.

Input: Command Activity TBD. Input at the external IST(s) and/or internal EOC user stations TBD, to invoke the Table Load Builder utility.

Output: Event messages in response to user input. Table load contents to include appropriate time tags and load mnemonics.

Expected Results: Successful build of Table load contents TBD. Table load contents available.

Test Case 19 EOC Ingest of Table Load Contents from External IST(s) Location(s)

This test verifies requirements related to the EOC capability to receive table load contents from external IST locations TBD.

Input: Table load contents available at IST locations TBD. Input/coordination at the EOC workstation TBD to invoke the transfer of table load contents to the EOC.

Output: Event messages in response to user inputs.

Expected Results: Successful transfer of the Table load contents to the appropriate EOC directory. Table load contents available to authorized users.

Test Case 20 EOC Validation of Table Load Content ID

This test verifies requirements related to the EOC capability to support the validation of table load content ID.

Input: Table load contents (TBD) available at the EOC for content ID validation. Input at the EOC workstation TBD to invoke the required utility for validation of load content ID.

Output: Event messages in response to user input. Report of content ID validation.

Expected Results: Successful validation of load content ID. Error/alarm messages for invalid load content ID.

Test Case 21 EOC Generation of Uplink Load for Table Contents

This test verifies requirements related to the EOC capability to support uplink load generation for table load contents received for IST locations TBD.

Input: Table load contents with valid load contents ID. Input at the EOC workstation to invoke the required utility to generate the uplink load.

Output: Event messages in response to user inputs. Report of uplink load generation results.

Expected Results: Successful generation of uplink load for table contents. Uplink load availability to all authorized users.

Test Case 22 ICC Receipt of IST Instrument Command Request

This test verifies requirements related to the EOC capability to support the ICC receipt of a instrument command request from the associated IST PI/TL.

Input: Input at the PI/TL IST to invoke the tools necessary for generation and transfer of request to the ICC.

Output: Event messages in response to user input.

Expected Results: Successful transfer of instrument command request from IST to ICC.

5.1.26 FOS Spacecraft Commanding Thread (T14-06)

5.1.26.1 Thread Objectives

This objective of this thread is to identify the FOS Spacecraft Commanding Subsystem (CMD) functionality provided in TRMM (Rel A) that provides the capability to uplink spacecraft and instrument commands to the AM-1 spacecraft. Specifically, this thread identifies the following Spacecraft Command functionality as it relates to the FOS TRMM (Rel A) Builds:

- Command Authority.
- EOC - EDOS Command Interface Connectivity.
- Flight Operations Ground Script Control.
- Manual (Real-Time) Spacecraft Commanding.
- EOC Command Transmission & Receipt Verification.

5.1.26.2 Thread Test Description

For a valid Flight Operations Ground Script and corresponding command loads, the Spacecraft Commanding Subsystem functionality provides for the execution of the ground script and uplink of command loads.

Spacecraft Commanding Thread testing is initiated by verifying the FOS User Interface (FUI) Subsystem supports the assignment of Command Activity Controller (CAC) privileges within the EOC. This phase of the testing verifies that only one (1) FOT workstation is assigned CAC privileges at any given time and the capability exist to assign CAC privileges to other EOC workstations TBD, without violation of the single point of commanding requirement for the EOC.

Following the completion of Command Authority testing, the test conductor will proceed to verify the EOC capability to support EOC -EDOS interface connectivity. Testing will verify test message 'handshakes" transferred between the EOC and EDOS testbed to confirm command communications capability.

Following successful completion of EOC-EDOS Interface testing, the test conductor will proceed to verify the EOC capability to support the execution of an operations ground script and the uplink of the corresponding command loads. This phase of the testing verifies the ground script execution can be initiated, suspended and resumed at the direction of the FOT user with CAC privileges. As the ground script executes, testing will verify processing, generation and transmission of the associated command loads. The command loads uplinked will include various Absolute Time Command (ATC) loads, Relative Time Command (RTC) loads, Instrument Microprocessor loads and Table loads TBD.

Following the completion of Ground Script Control testing, the test conductor will proceed to verify Manual (Real-Time) Commanding capabilities of the EOC. Testing will verify the capability at the FOT CAC workstation to input commands TBD. The real-time commands will be processed, validated, the bit pattern generated and transmitted for uplink to the spacecraft.

The final phase of Spacecraft Commanding will verify the capability of the EOC to support command transmission requirements related to transmission rates and transmission modes. Transmission requirements related to rates will be verified to confirm the EOC support command transmission rates of 1 kbps, 2 kbps, 10 kbps and 125 bps. Transmission requirements related to modes will be verified to confirm the EOC supports the following command configurations: Space Network (SN), Ground Network (GN), Deep Space Network (DSN) and Wallops Orbital Tracking Station (WOTS) (Note, not all command rates and or modes may be supported in FOS TRMM (Rel A)).

All commands transmitted from the EOC will be verified for proper receipt at the AM-1 spacecraft/testbed.

Dependencies: (If Applicable)

In order to verify the objectives of the spacecraft commanding thread, the following interface and functional capabilities are required:

- Command Management Thread
- Data Management Thread
- User Interface Thread
- Telemetry Thread
- EOC FOS - EDOS Interface

Test Support Requirements:

- Hardware:
  - TBS
- Software:
  - TBS
- Data:
  - TBS

5.1.26.2.1 Sequence 1 - EOC Command Authority

The following series of tests verify the EOC capability to assign Command Activity Controller (CAC) privileges to any one FOT workstation at any one time. This sequence of test cases will verify the capability to request command capabilities from an EOC workstation, request approval or denial and the reassignment of CAC privileges without violating single point of commanding requirements for the EOC.

<u>Test Case 1</u> <u>EOC FOT Workstation Command Authorization Request & Approval</u>

This test verifies the EOC capability to support a request for CAC privileges from any given EOC workstation.

Input: STOL input from EOC workstations TBD to request Command Activity Controller (CAC) privilege.

Output: Event message of current CAC assignment. Event message for approval and/or denial of request.

Expected Results: Successful assignment of CAC privileges to any one EOC workstation at a given time.

Test Case 2 EOC FOT Workstation Command Authorization Denied

This test verifies the EOC capability to deny an EOC user request for CAC privileges based on the user being unauthorized and/or the present assignment of CAC privileges to another user.

Input: STOL input at an EOC workstation TBD to request CAC privileges.

Output: Event message at the EOC workstation confirming insufficient authorization for CAC privileges and/or CAC privileges already assigned to another workstation.

Expected Results: Verification that unauthorized users are unable to obtain CAC privilege. Verification that the user workstation with present CAC privilege is able to "block" or deny CAC request from another user workstation.

5.1.26.2.2 Sequence 2 - EOC - EDOS Command Interface Connectivity

The following series of tests verifies the EOC - EDOS command interface and EDOS capability to support EOC uplink of spacecraft commands in CCSDS packet format.

Test Case 1 EOC Transfer of Communication Test Message to EDOS

This test verifies the EOC - EDOS communications connectivity via test messages transferred across the interface.

Input: Input at the EOC workstation TBD to invoke the generation and transfer of a communications test message to the EDOS facility.

Output: Event messages reporting generation and transfer of test message to the EDOS facility.

Expected Results: Successful transfer of the test message and an acknowledgment message received at the EOC from EDOS.

Test Case 2 EOC Command Transmission to EDOS in CCSDS Packet Format

This test verifies the capability of the EOC to transmit spacecraft commands to the EDOS facility in CCSDS packet format.

Input: Valid commands TBD available for uplink. Input at the EOC CAC workstation to invoke the transmission of commands to the EDOS in CCSDS packet format.

Output: Transmission of the commands from the EOC to EDOS. Event messages verifying commands transmitted. Command history logging.

Expected Results: Successful receipt of CCSDS format commands at EDOS.

5.1.26.2.3 Sequence 3 - EOC Flight Operations Ground Script Control

The following series of tests verify the capability to control spacecraft commanding and ground system directives via the operations ground script. The test cases in this sequence will verify the capability to configure ground system equipment and uplink commands to the spacecraft via ground script execution.

Test Case 1 Ground Script Execution Initialization

This test verifies that the EOC FOT user with CAC privileges is capable of initializing the execution of a flight operations ground script.

Input: Valid Flight Operations Ground Script. STOL input at the EOC CAC workstation to invoke the execution of the ground script.

Output: Event messages verifying the initialization of ground script. Event messages of time ordered directives being executed within the script. Ground script prompt messages for CAC input.

Expected Results: Successful execution of the ground script to include command load.

Test Case 2 Ground Script Suspend/Resume and Jump Capabilities

This test verifies the EOC capability to suspend an executing ground script, resume a suspended ground script and jump to directives out of order within the script.

Input: Valid ground script executing at the EOC. Input at the EOC CAC workstation to suspend the script, resume the script and jump (skip) selected directives TBD.

Output: Event message notifying the CAC that the script has been suspended. Event message notifying the CAC that script execution has resumed. Event message notifying the CAC that the script has jumped to a directive out of order and is executing starting with the CAC selected directive.

Expected Results: Upon suspend, any further execution of script directives should suspend. Successful resume of script execution and successful use of the CAC jump request.

Test Case 3 Uplink of AM-1 Spacecraft Critical Commands via Ground Script Execution

This test verifies the EOC capability to support critical command uplink via ground script execution.

Input: Valid ground script and associated critical commands. Input at the CAC workstation to invoke the execution of the ground script.

Output: Event message prompting CAC input of "ALLOW" or "CANCEL" prior to ground script execution of critical command uplink.

Expected Results: "ALLOW" response at the CAC workstation should execute the uplink of critical commands to EDOS in CCSDS format. "CANCEL" response at CAC workstation should cancel the uplink of the critical commands and jump to a designated directive in the ground script.

Test Case 4 Uplink of ATC Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an ATC load.

Input: Valid Ground Script and associated ATC load(s). Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for that portion of the script that executes uplink of the ATC load.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated ATC load in CCSDS packet.

Test Case 5 Uplink of RTC Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an RTC load.

Input: Valid Ground Script and associated RTC load(s). Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for that portion of the script that executes uplink of the RTC load.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated RTC load in CCSDS packet.

Test Case 6 Uplink of CERES Instrument Microprocessor Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an CERES inst. microprocessor load.

Input: Valid Ground Script and associated CERES inst. microprocessor load. Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for the portion of the ground

script that executes uplink of the CERES inst. microprocessor load. Uplink status message generated to CERES IST.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated CERES inst. microprocessor load in CCSDS packet format.

Test Case 7 Uplink of MISR Instrument Microprocessor Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an MISR inst. microprocessor load.

Input: Valid Ground Script and associated MISR inst. microprocessor load. Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for the portion of the ground script that executes uplink of the MISR inst. microprocessor load. Uplink status message generated to MISR IST.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated MISR inst. microprocessor load in CCSDS packet format.

Test Case 8 Uplink of MODIS Instrument Microprocessor Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an MODIS inst. microprocessor load.

Input: Valid Ground Script and associated MODIS inst. microprocessor load. Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for the portion of the ground script that executes uplink of the MODIS inst. microprocessor load. Uplink status message generated to MODIS IST.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated MODIS inst. microprocessor load in CCSDS packet format.

Test Case 9 Uplink of MOPITT Instrument Microprocessor Load via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an MOPITT inst. microprocessor load.

Input: Valid Ground Script and associated MOPITT inst. microprocessor load. Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for the portion of the ground script that executes uplink of the MOPITT inst. microprocessor load. Uplink status message generated to MOPITT IST.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated MOPITT inst. microprocessor load in CCSDS packet format.

Test Case 10 Uplink of Table Load(s) via Ground Script Execution

This test verifies the EOC capability to execute a flight operations ground script that includes directives for the uplink of an Table load.

Input: Valid Ground Script and associated Table load(s) TBD. Input at the EOC CAC workstation to execute to ground script.

Output: Event messages verifying the step-by-step execution of ground script directives. Event messages for the portion of the ground script that executes uplink of the Table load.

Expected Results: Successful execution of the ground script, to include successful uplink of the associated Table load in CCSDS packet format.

Test Case 11 EOC Uplink of Emergency/Contingency Instrument Commands

This test verifies the EOC capability to execute the uplink of emergency instrument commands that reside on the system. Testing will verify the EOC provides instrument command notification messages to the associated ICC during uplink of the emergency command activity.

Input: Valid emergency/contingency commands available for uplink. Input at the EOC CAC workstation to invoke the tools necessary for uplink execution.

Output: Event messages in response to user input.

Expected Results: Successful uplink of commands to the spacecraft and associated instrument. Successful transfer of instrument command notification messages to the ICC from the EOC.

5.1.26.2.4 Sequence 4 - EOC CAC Manual (Real-Time) Spacecraft Commanding

The following series of tests verify the EOC capability to allow manual command input at the CAC user workstation. Testing will verify the EOC capability to process the manual commands, validate, generate the associated bit pattern and transmit the commands to the AM-1 spacecraft via EDOS.

Test Case 1 — EOC Real-Time Discrete Commanding of Spacecraft Control Computer

This test verifies the EOC capability to support real-time commanding of the AM-1 Spacecraft Control Computer (SCC). This test case will verify the capability to enter commands at the EOC CAC workstation to change the mode of the AM-1 SCC (e.g. Initialization, Standby, Active, Operational, Halt).

Input: Valid discrete SCC mode commands. Input at the CAC workstation to uplink SCC discrete commands.

Output: Event messages at the CAC workstation verifying command processing, command validation, generation and transmission of the command. History logging of command events.

Expected Results: Successful transmission of SCC discrete commands to EDOS in CCSDS packet format.

Test Case 2 — EOC Manual Uplink of CERES IST Generated Instrument Commands/Loads

This test verifies EOC capability to support the real-time uplink of CERES instrument commands/loads received from the CERES IST.

Input: CERES instrument commands/loads TBD available at EOC for uplink. Input at the CAC workstation for manual uplink of CERES commands/loads.

Output: Event message at the CAC workstation verifying command processing, command validation, generation and transmission of commands/loads to EDOS. History logging of command events.

Expected Results: Successful transmission of CERES instrument commands/loads to EDOS in CCSDS packet format. Instrument command notification messages transferred to the ICC/IST.

Test Case 3 — EOC Manual Uplink of MISR IST Generated Instrument Commands/Loads

This test verifies EOC capability to support the real-time uplink of MISR instrument commands/loads received from the MISR IST.

Input: MISR instrument commands/loads TBD available for uplink. Input at the CAC workstation for manual uplink of MISR commands.

Output: Event message at the CAC workstation verifying command processing, command validation, generation and transmission of command to EDOS. History logging of command events.

Expected Results: Successful transmission of MISR instrument commands/loads to EDOS in CCSDS packet format. Instrument command notification messages transferred to the ICC/IST.

Test Case 4 — EOC Manual Uplink of MODIS IST Generated Instrument Commands/Loads

This test verifies EOC capability to support the real-time uplink of MODIS instrument commands received from the MODIS IST.

Input: MODIS instrument commands TBD available at EOC for uplink. Input at the CAC workstation for manual uplink of MODIS commands.

Output: Event message at the CAC workstation verifying command processing, command validation, generation and transmission of command to EDOS. History logging of command events.

Expected Results: Successful transmission of MODIS instrument commands to EDOS in CCSDS packet format. Instrument command notification messages transferred to the ICC/IST.

Test Case 5 — EOC Manual Uplink of MOPITT IST Generated Instrument Commands/Loads

This test verifies EOC capability to support the real-time uplink of MOPITT instrument commands/loads received from the MOPITT IST.

Input: MOPITT instrument commands/loads TBD available for uplink. Input at the CAC workstation for manual uplink of MOPITT commands.

Output: Event message at the CAC workstation verifying command processing, command validation, generation and transmission of command to EDOS. History logging of command events.

Expected Results: Successful transmission of MOPITT instrument commands/loads to EDOS in CCSDS packet format. Instrument command notification messages transferred to the ICC/IST.

Test Case 6 — EOC Manual Uplink of ASTER ICC/IST Generated Instrument Commands/Loads

This test verifies the EOC capability to support the real-time uplink of ASTER instrument commands/loads received from the ASTER ICC.

Input: ASTER instrument commands/loads TBD. Input at the CAC workstation to invoke manual uplink.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission of commands. History logging of command events.

Expected Results: Successful transmission of ASTER instrument commands/loads to EDOS in CCSDS packet format. Instrument command notification messages transferred to the ICC/IST.

Test Case 7 EOC Manual Uplink of AM-1 Spacecraft Critical Commands

This test verifies the EOC capability to support the real-time uplink of critical commands manually entered at the CAC workstation.

Input: AM-1 Spacecraft critical commands TBD. Input at the CAC workstation to invoke command uplink process.

Output: Event message prompt of CAC to "ALLOW" or "CANCEL" the uplink of a critical command. Command history logging.

Expected Results: Successful processing, validation and generation of the critical commands. Successful uplink upon CAC "ALLOW" input at the workstation. Uplink cancellation upon CAC "CANCEL" input at the workstation.

Test Case 8 EOC Manual Uplink of Absolute Time Command Load

This test verifies the EOC capability to support real-time uplink of an ATC load via manual input at the EOC CAC workstation.

Input: Valid ATC Load. Input at the CAC workstation to invoke load uplink.

Output: Event message verifying command processing, validation, generation and transmission.

Expected Results; Successful transmission of the ATC load to EDOS in CCSDS packet format.

5.1.26.2.5 Sequence 5 - EOC Command Transmission & Receipt Verification

The following series of tests verify the EOC capability to support command transmission at required rates and modes as well as command receipt verification. Specifically, this sequence of test cases verify the capability to transmit commands to EDOS for uplink to the spacecraft at rates of 1 kbps, 2 kbps, 10 kbps and 125 kbps. In addition, these test cases verify the capability to transmit commands to EDOS for uplink to the spacecraft via the SN, GN, DSN and WOTS. Spacecraft receipt verification is also exercised during this sequence.

Test Case 1 EOC Command Transmission to EDOS for Uplink at 1 kbps

This test verifies the EOC capability to support command transmission to EDOS for uplink to the AM-1 spacecraft at a transmit rate of 1 kbps.

Input: Valid spacecraft/instrument commands TBD. Input at the EOC CAC workstation to invoke command uplink at a 1 kbps rate.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS at 1 kbps rate. EOC and EDOS capability to verify rate.

Test Case 2 EOC Command Transmission to EDOS for Uplink at 2 kbps

This test verifies the EOC capability to support command transmission to EDOS for uplink to the AM-1 spacecraft at a transmit rate of 2 kbps.

Input: Valid spacecraft/instrument commands TBD. Input at the EOC CAC workstation to invoke command uplink at a 2 kbps rate.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS at 2 kbps rate. EOC and EDOS capability to verify rate.

Test Case 3 EOC Command Transmission to EDOS for Uplink at 10 kbps

This test verifies the EOC capability to support command transmission to EDOS for uplink to the AM-1 spacecraft at a transmit rate of 10 kbps.

Input: Valid spacecraft/instrument commands TBD. Input at the EOC CAC workstation to invoke command uplink at a 10 kbps rate.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS at 10 kbps rate. EOC and EDOS capability to verify rate.

Test Case 4 EOC Command Transmission to EDOS for Uplink at 125 bps

This test verifies the EOC capability to support command transmission to EDOS for uplink to the AM-1 spacecraft at a transmit rate of 125 bps.

Input: Valid spacecraft/instrument commands TBD. Input at the EOC CAC workstation to invoke command uplink at a 125 bps rate.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS at 125 bps rate. EOC and EDOS capability to verify rate.

Test Case 5 EOC Command Transmission to EDOS for Uplink Using the NASA Space Network (SN)

This test verifies the EOC capability to transmit commands to the AM-1 spacecraft using the NASA Space Network.

Input: Valid spacecraft/instrument commands TBD. Input at the CAC workstation to invoke command transmission in SN mode.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS in CCSDS format and SN format if different from CCSDS. EOC and EDOS capability to verify transmission mode of SN.

Test Case 6 EOC Command Transmission to EDOS for Uplink Using the NASA Ground Network (GN)

This test verifies the EOC capability to transmit commands to the AM-1 spacecraft using the NASA Ground Network.

Input: Valid spacecraft/instrument commands TBD. Input at the CAC workstation to invoke command transmission in GN mode.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS in CCSDS format and GN format if different from CCSDS. EOC and EDOS capability to verify transmission mode of GN.

Test Case 7 EOC Command Transmission to EDOS for Uplink Using the NASA Deep Space Network (DSN)

This test verifies the EOC capability to transmit commands to the AM-1 spacecraft using the NASA Deep Space Network (DSN).

Input: Valid spacecraft/instrument commands TBD. Input at the CAC workstation to invoke command transmission in DSN mode.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS in CCSDS format and DSN format if different from CCSDS. EOC and EDOS capability to verify transmission mode of DSN.

Test Case 8 EOC Command Transmission to EDOS for Uplink Using the NASA Wallops Orbital Tracking Station (WOTS)

This test verifies the EOC capability to transmit commands to the AM-1 spacecraft using the NASA WOTS.

Input: Valid spacecraft/instrument commands TBD. Input at the CAC workstation to invoke command transmission in WOTS mode.

Output: Event message at the CAC workstation verifying command processing, validation, generation and transmission. Command history logging.

Expected Results: Successful transmission of commands to EDOS in CCSDS format and WOTS format if different from CCSDS. EOC and EDOS capability to verify transmission mode of WOTS.

Test Case 9 EOC Command Receipt Verification

This test verifies the EOC capability to support command receipt verification. Testing verifies that the spacecraft received commands TBD successfully in respect to how the commands appeared at the EOC prior to uplink transmission.

Input: Valid commands TBD available for uplink. Input at the CAC workstation to invoke uplink of the commands. Telemetry receipt at the EOC.

Output: Event messages verifying command processing, validation, generation and transmission of commands. Command history logging.

Expected Results: Telemetry verification of command receipt. Analysis of CLCW in the return telemetry stream for command receipt verification.

5.1.27 FOS Telemetry Processing Thread (T14-07)

5.1.27.1 Thread Objectives

The objectives of the telemetry processing thread are as follows:

- Provide capability to receive instrument/spacecraft housekeeping and health and safety CCSDS telemetry packets by the FOS.
- Provide capability to broadcast instrument/spacecraft data housekeeping and health & safety telemetry packets to EOC and IST user stations.
- Provide the capability to perform instrument/spacecraft housekeeping and health & safety telemetry decommutation including EU conversion, limits checking, delta limits checking.
- Provide the capability to perform instrument/spacecraft housekeeping and health & safety data packet quality checking.
- Provide the capability to perform IST instrument/spacecraft housekeeping and health & safety telemetry mirror mode processing.
- Provide the user interface display functionality for instrument/spacecraft and health & safety telemetry.

5.1.27.2 Thread Test Description

Several EOC user stations will be powered up and configured to the AM-1 spacecraft configuration in support of housekeeping telemetry processing; other EOC user stations will be

signed on to support different AM-1 configurations. Following system initialization and configuration, AM-1 spacecraft/instrument housekeeping telemetry data will be transmitted via EDOS for subsequent receipt, broadcast, and processing by various EOC user station groups. User interface room/page displays of telemetry parameter values, associated flags, telemetry event messages, archive data directories and other operational information will be used as a basis for both viewing operational activity and performing test analysis. Following verification of housekeeping telemetry processing, health & safety telemetry processing will be verified using the same steps outlined above. Finally IST user stations will be configured to accept specific telemetry from the FOS for mirror mode processing; packets will be broadcast and telemetry processing will be verified using the IST sets of user stations.

Dependencies:

In order to verify the objectives of the telemetry processing thread, the following interface and functional capabilities are required:

- EDOS-ECOM
- User Interface
- Data Base

Test Support Requirements

- Hardware:
  - EOC user stations
  - T&C Processor
  - FOS LAN
- Software:
  - ECOM
  - EDOS
  - AM-1 Data simulator
- Data:
  - AM-1 spacecraft/instrument housekeeping data

5.1.27.2.1 Sequence 1 - Housekeeping Data Receipt

This sequence of tests focuses on the verification of housekeeping telemetry data reception and broadcast to configured EOC user stations.

<u>Test Case 1</u> <u>Housekeeping - Telemetry Packet Receipt</u>

This test verifies correct receipt of EDOS-provided AM-1 housekeeping CCSDS telemetry data packets and subsequent broadcast of telemetry data to properly configured EOC user stations.

Input:	Housekeeping CCSDS packet transmission at the EDOS-provided real-time rates. User directives to set up FOS configuration.
Output:	Housekeeping packet dumps as received at the T&C processor, and EOC user stations. Event message displays.
Expected Results:	Event messages stating packet receipt at the T&C; Receipt of data at the EOC user stations and event messages stating receipt at configured user stations. Dumps of housekeeping CCSDS packets at T&C, EOC user stations in compliance with defined CCSDS packet format.

Test Case 2 Housekeeping - Incomplete Packet Processing

This test verifies FOS' rejection and subsequent event notification of incomplete AM-1 housekeeping telemetry packet receipt at the EOC set of configured user stations.

Input:	Scripted housekeeping data at EDOS-provided transmission rates. User directives to set up FOS configuration.
Output:	Event message and various telemetry room/page displays at individual EOC user stations; telemetry archive directory.
Expected Results:	Event messages stating incomplete packet receipt at those EOC user stations previously configured to accept data. Telemetry data not processed (i.e., decommutated, EU converted, displayed, etc.).

Test Case 3 Housekeeping - WOTS Data Receipt

This test verifies the capability to accept housekeeping telemetry data originating from the WOTS.

Input:	Scripted housekeeping telemetry data at EDOS-provided transmission rates and all containing source ID from the WOTS downlink source. User directives to set up FOS configuration.
Output:	Event message and various telemetry room/page displays at EOC user stations.
Expected Results:	Event messages stating valid receipt of EDOS-provided housekeeping telemetry from WOTS. Housekeeping telemetry processed when received; parameter values/flags match those scripted.

Test Case 4 Housekeeping - DSN Data Receipt

This test verifies the capability to accept housekeeping telemetry data originating from the DSN.

Input:	Scripted housekeeping telemetry data at EDOS-provided transmission rates and all containing source ID from the DSN downlink source. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating valid receipt of EDOS-provided spacecraft/housekeeping telemetry from DSN. Housekeeping telemetry processed when received; parameter values/flags match those scripted.

Test Case 5 Housekeeping - Packet Receipt With Unknown Source ID

This test verifies the rejection and subsequent event notification of housekeeping telemetry data packets with invalid source ID at the EOC set of configured user stations.

Input: Scripted housekeeping telemetry data with invalid source ID at EDOS-provided transmission rates. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at individual EOC user stations; telemetry archive directory.

Expected Results: Event messages stating unknown packet source ID; Telemetry data not processed (i.e., decommutated, EU converted, displayed, etc.).

Test Case 6 Housekeeping - Reject Bad Quality Packets

This test verifies the ability to reject and report housekeeping telemetry packets previously flag as questionable quality by EDOS.

Input: Scripted housekeeping telemetry data packets marked as questionable quality; transmitted from EDOS at EDOS-provided rates. User directives to set up FOS configuration.

Output: Event message and telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating receipt of telemetry packets containing questionable quality. Decommutation of all housekeeping parameters within packets marked with questionable quality not performed. All housekeeping telemetry parameters resident in questionable quality packet are flagged accordingly on telemetry display pages.

5.1.27.2.2 Sequence 2 - Mark Static Housekeeping Telemetry

This sequence of tests verifies the ability to mark housekeeping telemetry parameters static upon varying "no data" or "data dropout" conditions.

Test Case 1 Housekeeping - Static Condition Upon Data Dropout

This test verifies the ability to flag individual housekeeping telemetry parameters as "static" following a set period of non-decommutation activity (i.e., data dropout) , and to report the condition via event messages.

Input: Housekeeping telemetry data dropout via EDOS. User directives to set up FOS configuration.

Output: Event messages, telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating data dropout. Those displayed housekeeping parameters not decommutated for a period of TBD seconds are marked as "static".

Test Case 2 Housekeeping - Static Condition Upon Bad Quality Packet Receipt

This test verifies the ability to flag individual housekeeping telemetry parameters as "static" following a set period of bad quality housekeeping data receipt, and to report the condition via event messages.

Input: Scripted housekeeping telemetry packets marked as "bad quality" by EDOS. User directives to set up FOS configuration.

Output: Event messages, telemetry room/display pages at EOC user stations.

Expected Results: Event messages stating bad quality data receipt. Those displayed housekeeping parameters not decommutated for a period of TBD seconds are marked as "static".

5.1.27.2.3 Sequence 3 - Housekeeping Decommutation/Conversion

This sequence of tests verifies the ability to decommutate/EU convert broadcasted housekeeping telemetry data according to data base defined EU definitions and to report any conversion errors caused during the EU conversion process.

Test Case 1 Housekeeping - EU Conversion using Polys/linear interpolations

This test verifies the ability to decommutate housekeeping telemetry parameters according to individual EU polynomial conversion curves or individual linear interpolation definitions containing one or more point pairs.

Input: Scripted housekeeping telemetry data provided and transmitted from EDOS at EDOS-provided rates. Previously defined room/page telemetry displays.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Housekeeping telemetry parameters decommutated according to the data base defined polynomial or linear interpolation. Proper display of decommutated value (i.e., numeric or text), and data base defined units (i.e., amps, volts, degrees, etc.). Individual housekeeping telemetry values and units displayable via user request.

Test Case 2 Housekeeping - EU Conversion Selection Based on Parameter Key

This test verifies the ability to decommutate housekeeping telemetry parameters based on data base defined parameter keys.

Input: Scripted housekeeping data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Housekeeping telemetry parameters decommutated according to the data base defined key parameter. Proper display of decommutated value (i.e., numeric or text), and data base defined units (i.e., amps, volts, degrees, etc.).

Test Case 3 Housekeeping - EU Conversion Error

This test verifies the ability to flag individual housekeeping telemetry data parameters as "conversion error" when EU conversion results in out of range conditions.

Input: Scripted housekeeping telemetry data resulting in conversion error conditions; transmitted from EDOS at EDOS-provided rates.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating conversion error condition at configured EOC user stations. Those housekeeping telemetry parameters whose EU conversion results in the conversion error condition are appropriately flagged on telemetry room/page displays.

5.1.27.2.4 Sequence 4 - Housekeeping Limits Checking

This sequence of tests verifies the ability to limits check broadcasted housekeeping telemetry data according to data base defined EU definitions and to report any limits violations during periods of limits processing activity.

Test Case 1 Housekeeping - High/Low, Red/Yellow Limits

This test verifies the ability to display and report on housekeeping telemetry parameters whose raw or EU current values are out of limits based on data-based defined red/yellow, high/low range limit definitions.

Input: Scripted housekeeping telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Housekeeping telemetry parameters with limit values out of range are reported as out of limits on room/page displays and match the individual data base defined limit definition (i.e., out of limits red

high, red low, yellow high or yellow low). Event message/alarm message display stating out of limits conditions. Individual housekeeping telemetry parameter limit values (high/low and red/yellow) displayable upon user request.

Test Case 2 Housekeeping - Limit Set Selection Based on Parameter Key

This test verifies the ability to process housekeeping telemetry parameter limits based on the limit set invoked via another parameter value according to data base parameter limit definitions.

Input: Scripted housekeeping telemetry transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with defined sets of limit ranges.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Housekeeping telemetry parameters are limits processed based on limit set chosen by parameter key; parameter key values and limits display matches the operational limit set, both before and after limit switch activity. Individual housekeeping telemetry parameter active limit set displayable upon user request.

Test Case 3 Housekeeping - Limits Processing Based on Bad Quality Data

This test verifies inhibition of limit processing on housekeeping telemetry data parameters marked as "questionable quality".

Input: Scripted housekeeping telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with data base defined limits (both raw and EU).

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: All housekeeping telemetry parameters marked as "questionable quality" are not limit checked and don't display limit flags (unless limits condition occurs first).

Test Case 4 Housekeeping - Delta Limits

This test verifies the ability to display and report on housekeeping telemetry parameters whose successive raw values result in a delta limit error according to data base defined delta limits information.

Input: Scripted housekeeping telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with data base defined delta limits.

Output: Delta limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Housekeeping telemetry parameters whose successive raw value difference exceeds data base defined delta limit definitions are reported on room/page displays and delta limit event messages are generated. Individual parameter delta limit definitions displayable upon user request and match data base definitions.

Test Case 5 Housekeeping - Turning Limits Reporting On/Off

This test verifies the ability to turn limits reporting on/off at individual EOC user stations.

Input: Scripted housekeeping telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Housekeeping telemetry parameters with limit values out of range are reported as out of limits on room/page displays only, but limit event messages do not appear on event page. Turning limits reporting on/off does not affect limits processing and does not affect limits processing and reporting at other user stations.

5.1.27.2.5 Sequence 5 - Health & Safety Data Receipt

This sequence of tests focuses on the verification of health & safety telemetry data reception and broadcast to configured EOC user stations.

Test Case 1 Health & Safety - Telemetry Packet Receipt

This test verifies correct receipt of EDOS-provided AM-1 health & safety CCSDS telemetry data packets and subsequent broadcast of telemetry data to properly configured EOC user stations.

Input: Health & safety CCSDS packet transmission at the EDOS-provided real-time rates. User directives to set up FOS configuration.

Output: Health & safety packet dumps as received at the T&C processor, and EOC user stations. Event message displays.

Expected Results: Event messages stating packet receipt at the T&C; Receipt of data at the EOC user stations and event messages stating receipt at configured user stations. Dumps of health & safety CCSDS packets at T&C, EOC user stations in compliance with defined CCSDS packet format.

Test Case 2 Health & Safety - Incomplete Packet Processing

This test verifies FOS' rejection and subsequent event notification of incomplete AM-1 health & safety telemetry packet receipt at the EOC set of configured user stations.

Input: Scripted health & safety data at EDOS-provided transmission rates. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at individual EOC user stations; telemetry archive directory.

Expected Results: Event messages stating incomplete packet receipt at those EOC user stations previously configured to accept data. Telemetry data not processed (i.e., decommutated, EU converted, displayed, etc.).

Test Case 3 Health & Safety - WOTS Data Receipt

This test verifies the capability to accept health & safety telemetry data originating from the WOTS.

Input: Scripted health & safety telemetry data at EDOS-provided transmission rates and all containing source ID from the WOTS downlink source. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating valid receipt of EDOS-provided health & safety telemetry from WOTS. Health & safety telemetry processed when received; parameter values/flags match those scripted.

Test Case 4 Health & Safety - DSN Data Receipt

This test verifies the capability to accept health & safety telemetry data originating from the DSN.

Input: Scripted health & safety telemetry data at EDOS-provided transmission rates and all containing source ID from the DSN downlink source. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating valid receipt of EDOS-provided spacecraft/health & safety telemetry from DSN. Health & safety telemetry processed when received; parameter values/flags match those scripted.

Test Case 5 Health & Safety - Packet Receipt With Unknown Source ID

This test verifies the rejection and subsequent event notification of health & safety telemetry data packets with invalid source ID at the EOC set of configured user stations.

Input: Scripted health & safety telemetry data with invalid source ID at EDOS-provided transmission rates. User directives to set up FOS configuration.

Output: Event message and various telemetry room/page displays at individual EOC user stations; telemetry archive directory.

Expected Results: Event messages stating unknown packet source ID; Telemetry data not processed (i.e., decommutated, EU converted, displayed, etc.).

Test Case 6 Health & Safety - Reject Bad Quality Packets

This test verifies the ability to reject and report health & safety telemetry packets previously flag as questionable quality by EDOS.

Input: Scripted health & safety telemetry data packets marked as questionable quality; transmitted from EDOS at EDOS-provided rates. User directives to set up FOS configuration.

Output: Event message and telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating receipt of telemetry packets containing questionable quality. Decommutation of all health & safety parameters within packets marked with questionable quality not performed. All health & safety telemetry parameters resident in questionable quality packet are flagged accordingly on telemetry display pages.

5.1.27.2.6 Sequence 6 - Mark Static Health & Safety Telemetry

This sequence of tests verifies the ability to mark health & safety telemetry parameters static upon varying "no data" or "data dropout" conditions.

Test Case 1 Health & Safety - Static Condition Upon Data Dropout

This test verifies the ability to flag individual health & safety telemetry parameters as "static" following a set period of non-decommutation activity (i.e., data dropout) , and to report the condition via event messages.

Input: Health & safety telemetry data dropout via EDOS. User directives to set up FOS configuration.

Output: Event messages, telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating data dropout. Those displayed health & safety parameters not decommutated for a period of TBD seconds are marked as "static".

Test Case 2 Health & Safety - Static Condition Upon Bad Quality Packet Receipt

This test verifies the ability to flag individual health & safety telemetry parameters as "static" following a set period of bad quality health & safety data receipt, and to report the condition via event messages.

Input: Scripted health & safety telemetry packets marked as "bad quality" by EDOS. User directives to set up FOS configuration.

Output: Event messages, telemetry room/display pages at EOC user stations.

Expected Results: Event messages stating bad quality data receipt. Those displayed health & safety parameters not decommutated for a period of TBD seconds are marked as "static".

5.1.27.2.7 Sequence 7 - Health & Safety Decommutation/Conversion

This sequence of tests verifies the ability to decommutate/EU convert broadcasted health & safety telemetry data according to data base defined EU definitions and to report any conversion errors caused during the EU conversion process.

Test Case 1 Health & Safety - EU Conversion using Polys/linear interpolations

This test verifies the ability to decommutate health & safety telemetry parameters according to individual EU polynomial conversion curves or individual linear interpolation definitions containing one or more point pairs.

Input: Scripted health & safety telemetry data provided and transmitted from EDOS at EDOS-provided rates. Previously defined room/page telemetry displays.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Health & safety telemetry parameters decommutated according to the data base defined polynomial or linear interpolation. Proper display of decommutated value (i.e., numeric or text), and data base defined units (i.e., amps, volts, degrees, etc.). Individual health & safety telemetry values and units displayable via user request.

Test Case 2 Health & Safety EU Conversion Selection Based on Parameter Key

This test verifies the ability to decommutate health & safety telemetry parameters based on data base defined parameter keys.

Input: Scripted health & safety data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Health & safety telemetry parameters decommutated according to the data base defined key parameter. Proper display of decommutated value (i.e., numeric or text), and data base defined units (i.e., amps, volts, degrees, etc.).

Test Case 3 Health & Safety - EU Conversion Error

This test verifies the ability to flag individual health & safety telemetry data parameters as "conversion error" when EU conversion results in out of range conditions.

Input: Scripted health & safety telemetry data resulting in conversion error conditions; transmitted from EDOS at EDOS-provided rates.

Output: Event message, telemetry room/page displays at EOC user stations.

Expected Results: Event messages stating conversion error condition at configured EOC user stations. Those health & safety telemetry parameters whose EU conversion results in the conversion error condition are appropriately flagged on telemetry room/page displays.

5.1.27.2.8 Sequence 8 - Health & Safety Limits Checking

This sequence of tests verifies the ability to limits check broadcasted health & safety telemetry data according to data base defined EU definitions and to report any limits violations during periods of limits processing activity.

Test Case 1 Health & Safety - High/Low, Red/Yellow Limits

This test verifies the ability to display and report on health & safety telemetry parameters whose raw or EU current values are out of limits based on data-based defined red/yellow, high/low range limit definitions.

Input: Scripted health & safety telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing health & safety telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Health & safety telemetry parameters with limit values out of range are reported as out of limits on room/page displays and match the individual data base defined limit definition (i.e., out of limits red high, red low, yellow high or yellow low). Event message/alarm message display stating out of limits conditions. Individual health & safety telemetry parameter limit values (high/low and red/yellow) displayable upon user request.

Test Case 2 Health & Safety - Limit Set Selection Based on Parameter Key

This test verifies the ability to process health & safety telemetry parameter limits based on the limit set invoked via another parameter value according to data base parameter limit definitions.

Input: Scripted health & safety telemetry transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing health & safety telemetry parameters with defined sets of limit ranges.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Health & safety telemetry parameters are limits processed based on limit set chosen by parameter key; parameter key values and limits display matches the operational limit set, both before and after limit switch activity. Individual health & safety telemetry parameter active limit set displayable upon user request.

Test Case 3 Health & Safety - Limits Processing Based on Bad Quality Data

This test verifies inhibition of limit processing on health & safety telemetry data parameters marked as "questionable quality".

Input: Scripted health & safety telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing health & safety telemetry parameters with data base defined limits (both raw and EU).

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: All health & safety telemetry parameters marked as "questionable quality" are not limit checked and don't display limit flags (unless limits condition occurs first).

Test Case 4 Health & Safety - Delta Limits

This test verifies the ability to display and report on health & safety telemetry parameters whose successive raw values result in a delta limit error according to data base defined delta limits information.

Input: Scripted health & safety telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing health & safety telemetry parameters with data base defined delta limits.

Output: Delta limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Health & safety telemetry parameters whose successive raw value difference exceeds data base defined delta limit definitions are

reported on room/page displays and delta limit event messages are generated. Individual parameter delta limit definitions displayable upon user request and match data base definitions.

Test Case 5 — Health & Safety - Turning Limits Reporting On/Off

This test verifies the ability to turn limits reporting on/off at individual EOC user stations.

Input: Scripted health & safety telemetry data transmitted from EDOS at EDOS-provided rates. Previously defined telemetry room/page displays containing health & safety telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at EOC user stations.

Expected Results: Health & safety telemetry parameters with limit values out of range are reported as out of limits on room/page displays only, but limit event messages do not appear on event page. Turning limits reporting on/off does not affect limits processing and does not affect limits processing and reporting at other user stations.

5.1.27.2.9 Sequence 9 - IST Telemetry Mirror Mode Processing

This sequence of tests focuses on the verification of telemetry packet receipt and processing capabilities at the IST set of user stations.

Test Case 1 — IST - Housekeeping Telemetry Receipt

This test verifies correct receipt of FOS provided broadcast housekeeping telemetry and subsequent decommutation at the IST(s) using linear and polynomial interpolations.

Input: Previously scripted housekeeping data broadcast. User directives to set up FOS configuration.

Output: Housekeeping telemetry page display at IST user stations. Event message displays.

Expected Results: Event messages stating packet receipt at the IST(s). Telemetry decommutated values match data base defined definitions.

Test Case 2 — IST - Housekeeping Data - Static Condition Upon Data Dropout

This test verifies the ability to report static data conditions at the IST(s).

Input: Housekeeping telemetry data dropout. User directives to set up FOS configuration.

Output: Event messages, telemetry room/page displays at IST user stations.

Expected Results: Event messages stating data dropout. Those displayed housekeeping parameters not decommutated for a period of TBD seconds are marked as "static".

Test Case 3 IST - Housekeeping Data - High/Low, Red/Yellow Limits

This test verifies the ability to display and report on housekeeping telemetry parameters whose raw or EU current values are out of limits based on data-based defined red/yellow, high/low range limit definitions.

Input: Scripted housekeeping data broadcast from the FOS. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at IST user stations.

Expected Results: Housekeeping telemetry parameters with limit values out of range are reported as out of limits on room/page displays and match the individual data base defined limit definition (i.e., out of limits red high, red low, yellow high or yellow low). Event message/alarm message display stating out of limits conditions. Individual housekeeping telemetry parameter limit values (high/low and red/yellow) displayable upon user request.

Test Case 4 IST - Housekeeping Data - Delta Limits

This test verifies the ability to display and report on housekeeping telemetry parameters whose successive raw values result in a delta limit error according to data base defined delta limits information.

Input: Scripted housekeeping telemetry data broadcast from FOS. Previously defined telemetry room/page displays containing housekeeping telemetry parameters with data base defined delta limits.

Output: Delta limit event, alarm and status messages, telemetry display pages at IST user stations.

Expected Results: Housekeeping telemetry parameters whose successive raw value difference exceeds data base defined delta limit definitions are reported on room/page displays and delta limit event messages are generated. Individual parameter delta limit definitions displayable upon user request and match data base definitions.

Test Case 5 IST - Health & Safety Telemetry Receipt

This test verifies correct receipt of FOS provided broadcast health & safety telemetry and subsequent decommutation at the IST(s) using linear and polynomial interpolations.

Input: Previously scripted health & safety data broadcast. User directives to set up FOS configuration.

Output: Health & safety telemetry page display at IST user stations. Event message displays.

Expected Results: Event messages stating packet receipt at the IST(s). Telemetry decommutated values match data base defined definitions.

Test Case 6 IST - Health & Safety Data - Static Condition Upon Data Dropout

This test verifies the ability to report health & safety static data conditions at the IST(s).

Input: Health & safety telemetry data dropout. User directives to set up FOS configuration.

Output: Event messages, telemetry room/page displays at IST user stations.

Expected Results: Event messages stating data dropout. Those displayed health & safety parameters not decommutated for a period of TBD seconds are marked as "static".

Test Case 7 IST - Health & Safety Data - High/Low, Red/Yellow Limits

This test verifies the ability to display and report on health & safety telemetry parameters whose raw or EU current values are out of limits based on data-based defined red/yellow, high/low range limit definitions.

Input: Scripted health & safety data broadcast from the FOS. Previously defined telemetry room/page displays containing health & safety telemetry parameters with defined red/yellow, high/low range limit definitions.

Output: Limit event, alarm and status messages, telemetry display pages at IST user stations.

Expected Results: Health & safety telemetry parameters with limit values out of range are reported as out of limits on room/page displays and match the individual data base defined limit definition (i.e., out of limits red high, red low, yellow high or yellow low). Event message/alarm message display stating out of limits conditions. Individual health & safety telemetry parameter limit values (high/low and red/yellow) displayable upon user request.

Test Case 8 IST - Health & Safety Data - Delta Limits

This test verifies the ability to display and report on health & safety telemetry parameters whose successive raw values result in a delta limit error according to data base defined delta limits information.

Input: Scripted health & safety telemetry data broadcast from FOS. Previously defined telemetry room/page displays containing health & safety telemetry parameters with data base defined delta limits.

Output: Delta limit event, alarm and status messages, telemetry display pages at IST user stations.

Expected Results: Health & safety telemetry parameters whose successive raw value difference exceeds data base defined delta limit definitions are

reported on room/page displays and delta limit event messages are generated. Individual parameter delta limit definitions displayable upon user request and match data base definitions.

5.1.28 FOS Analysis Thread (T14-08)

5.1.28.1 Thread Objectives

The objectives of the analysis thread are as follows:

- Provide the capability to perform FOS on-line archiving of AM-1 historical housekeeping/health & safety telemetry.
- Provide the capability to perform AM-1 spacecraft/instrument analysis based on AM-1 spacecraft/instrument housekeeping and health & safety telemetry residing in the FOS on-line archive.
- Provide the capability to generate user reports (downlink order list reports, parameter out of limits reports) which contain historical AM-1 telemetry parameter values for specified time periods at EOC and IST user stations.
- Provide the capability to generate statistical reports (both system generated and customized) which provide minimum, maximum, mean, standard deviation parameter values for orbital, hourly, weekly, daily, monthly reports.
- Provide the capability to display user and statistical reports at EOC and IST user stations.
- Provide user report/statistical report queue management.

5.1.28.2 Thread Test Description

Specific EOC/IST workstations will be powered up and subsequently configured to the AM-1 spacecraft and operational mode configuration; specific locations will be signed on to another spacecraft configuration. Each AM-1 spacecraft defined telemetry format will be transmitted (only one stream at a time) via the EDOS/ECOM interface for receipt, archiving, and cataloging via the FOS Telemetry and Command (T&C) Processor. Once each data type is received at the T&C processor, verification of archiving and automatic cataloging will then be performed.

Specific user reports will be requested at various EOC and IST user stations. Report output, including display and print capabilities will then be verified.

Dependencies:

In order to verify the objectives of the analysis thread, the following interface and functional capabilities are required:

- User Interface
- Data Management
- [a] Telemetry

Test Support Requirements

- Hardware:
  - Several EOC workstations
  - Several IST workstations
  - FOS Data Archive
  - FOS LAN
- Software:
  - ECOM
  - EDOS
  - AM-1 data simulator software
- Data:
  - AM-1 spacecraft/instrument housekeeping data
  - AM-1 health & Safety data
  - AM-1 Instrument engineering data

5.1.28.2.1 Sequence 1- Data Archiving

This sequence verifies requirements related to the archival of AM-1 defined telemetry types as received from EDOS, and the subsequent cataloging of incoming telemetry with unique entry identifiers.

Test Case 1 — Housekeeping Telemetry Archival

This test case verifies the ability to archive spacecraft/Instrument housekeeping telemetry in the 1 and 16 kbps formats once received from EDOS/ECOM.

Input: AM-1 spacecraft housekeeping data in 1/16 kbps formats.

Output: Archive file dumps and catalog entries.

Expected Results: Archive file and catalog entries match data sent from EDOS. Catalog entries match data received in archive and are named according to time received and type of data.

Test Case 2 — Bad Quality Housekeeping Telemetry Archival

This test case verifies the ability to archive housekeeping telemetry data flagged as bad data quality by EDOS.

Input: AM-1 spacecraft and instrument housekeeping data in 1 and 16 kbps rates.

Output: Archive file dumps and catalog entry dumps.

Expected Results: All data received processed and archived despite bad quality status; archive files and catalog entries match data sent from EDOS.

Catalog entries match data received in archive and are named according to time received and type of data.

Test Case 3 — Health & Safety Telemetry Archival

This test case verifies the ability to archive health and safety telemetry in the 1 kbps format once received from EDOS.

Input: AM-1 health and safety data in 1 kbps format.

Output: Archive file dumps and catalog entries.

Expected Results: Archive file and catalog entries match data sent from EDOS. Catalog entries match data received in archive and are named according to time received and type of data.

Test Case 4 — Bad Quality Health & Safety Telemetry Archival

This test case verifies the ability to archive health & safety telemetry data flagged as bad data quality by EDOS.

Input: AM-1 health & safety data in 1 and 16 kbps rates.

Output: Archive file dumps and catalog entry dumps.

Expected Results: All data received processed and archived despite bad quality status; archive files and catalog entries match data sent from EDOS. Catalog entries match data received in archive and are named according to time received and type of data.

Test Case 5 — Telemetry Data Archiving Enable/Disable

This test verifies the ability to enable and disable telemetry archiving at the EOC.

Input: Spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS archive configuration.

Output: Event messages at the EOC; archive dumps. Archive file directories.

Expected Results: Telemetry archiving may be disabled/enabled for authorized EOC users. All archiving activity matches that for specified telemetry stream only. All other configurations' archiving function not affected by other string archiving enable/disable directives.

5.1.28.2.2 Sequence 2 - Automatic (Auto) Statistics (Stats) Generation

This sequence verifies those requirements relating to the generation and display of system generated (automatic) statistics for AM-1 telemetry data and subsequent reporting of the computed statistical information at the EOC and IST user stations.

Test Case 1 Auto Stats - EOC - All Data Available

This test verifies the ability to provide auto statistics reports (i.e., min, max, mean, standard deviation) for orbital, hourly, daily and monthly reports for AM-1 spacecraft/instrument telemetry parameters residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics display activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Report output of requested mnemonics identify auto statistics for the time period requested and statistics values coincide with scripted telemetry data.

Test Case 2 Auto Stats - EOC - No Data Available

This test verifies the ability to report on "no data" condition when attempting to report auto statistics for a time period when no automatic statistics were generated.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics reporting activity.

Output: Event message displays.

Expected Results: Event messages at EOC user interface event page stating auto statistics activity incomplete due to no data condition.

Test Case 3 Auto Stats - EOC - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during an auto stats generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics reporting activity.

Output: Event message display.

Expected Results: Event message at EOC user interface event page stating incorrect setup file entry; statistics generation reporting not begun until all setup file information is valid.

Test Case 4 Auto Stats - EOC - Data Flags

This test verifies the ability to provide auto statistics reports from telemetry data containing telemetry flags (i.e., limits, delta limits, static, bad quality).

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report display activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics and those statistic calculations coincide with scripted telemetry data value computations. Computation of parameter statistics with any value "out of limit" results in "out of limit" result. Computation of parameter statistics with any value "out of delta limit" results in "out of delta limit" result. Computations of parameter statistics do not include any parameter sample(s) flagged as bad quality or static.

Test Case 5 Auto Stats - IST - All Data Available

This test verifies the ability to provide auto statistics reports (i.e., min, max, mean, standard deviation) for AM-1 spacecraft/instrument telemetry parameters residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics display activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Report output of requested mnemonics identify auto statistics for the time period requested and statistics values coincide with scripted telemetry data.

Test Case 6 Auto Stats - IST - No Data Available

This test verifies the ability to report on "no data" condition when attempting to report auto statistics for a time period when no automatic statistics were generated.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics reporting activity.

Output: Event message displays.

Expected Results: Event messages at IST user interface event page stating auto statistics activity incomplete due to no data condition.

Test Case 7 Auto Stats - IST - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during an auto stats generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS statistics reporting activity.

Output: Event message display.

Expected Results: Event message at IST user interface event page stating incorrect setup file entry; statistics generation reporting not begun until all setup file information is valid.

Test Case 8 Auto Stats - IST - Data Flags

This test verifies the ability to provide auto statistics reports from telemetry data containing telemetry flags (i.e., limits, delta limits, static, bad quality).

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report display activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics and those statistic calculations coincide with scripted telemetry data value computations. Computation of parameter statistics with any value "out of limit" results in "out of limit" result. Computation of parameter statistics with any value "out of delta limit" results in "out of delta limit" result. Computations of parameter statistics do not include any parameter sample(s) flagged as bad quality or static.

5.1.28.2.3 \Sequence 3 - Downlink Order Lists

This sequence verifies those requirements relating to downlink order list generation of AM-1 spacecraft/instrument housekeeping and health & safety telemetry data and subsequent reporting of the downlink list information at the EOC and IST user stations.

Test Case 1 Downlink Order Lists - EOC - All Data Available

This test verifies the ability to produce downlink ordered lists (i.e., parameter values listed in order of downlink) over a specified time period from spacecraft/instrument telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Downlink ordered list values coincide with scripted telemetry data; parameters reported are in downlink order.

Test Case 2 Downlink Order List - EOC - Back Orbit From GSFC

This test verifies the ability to produce downlink ordered lists from spacecraft/instrument telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry - available at FOS on-line archive and at

GSFC DAAC. User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Downlink ordered list values coincide with scripted telemetry data; parameters reported are in downlink order. Data received from the GSFC DAAC is used for downlink ordered list generation.

Test Case 3 Downlink Order List - EOC - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce downlink ordered list based on data not available at the FOS on-line archive and GSFC DAAC.

Input: User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the EOC user stations; event message displays.

Expected Results: Event messages at EOC user interface event page stating selected downlink ordered list generation activity incomplete due to no data condition.

Test Case 4 Downlink Order List - EOC - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during a downlink order list generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS downlink ordered list generation activity.

Output: Event message display.

Expected Results: Event message at EOC user interface event page stating incorrect setup file entry; downlink ordered list generation process not begun until all setup file information is valid.

Test Case 5 Downlink Order List - IST - All Data Available

This test verifies the ability to produce downlink ordered lists (i.e., parameter values listed in order of downlink) over a specified time period from spacecraft/instrument telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Downlink ordered list values coincide with scripted telemetry data; parameters reported are in downlink order.

Test Case 6 Downlink Order List - IST - Back Orbit Data from GSFC DAAC

This test verifies the ability to produce downlink ordered lists from spacecraft/instrument telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry - available at FOS on-line archive and at GSFC DAAC. User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Downlink ordered list values coincide with scripted telemetry data; parameters reported are in downlink order. Data received from the GSFC DAAC is used for downlink ordered list generation.

Test Case 7 Downlink Order List - IST - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce downlink ordered list based on data not available at the FOS on-line archive and GSFC DAAC.

Input: User directives to set up FOS downlink ordered list generation activity.

Output: Report display at the IST user stations; event message displays.

Expected Results: Event messages at IST user interface event page stating selected downlink ordered list generation activity incomplete due to no data condition.

Test Case 8 Downlink Order List - IST - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during a downlink order list generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS downlink ordered list generation activity.

Output: Event message display.

Expected Results: Event message at IST user interface event page stating incorrect setup file entry; downlink ordered list generation process not begun until all setup file information is valid.

5.1.28.2.4 Sequence 4 - Parameter Out of Limits List Reports

This sequence verifies those requirements relating to parameter out of limits list generation of AM-1 spacecraft/instrument housekeeping and health & safety telemetry data and subsequent reporting of the out of limit list information at the EOC and IST user stations.

Test Case 1 Parameter Out of Limit List - EOC - All Data Available

This test verifies the ability to produce parameter out of limits lists (i.e., parameter mnemonic name and associated limits flag and time tag) over a specified time period from spacecraft/instrument telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS parameter out of limits list generation activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Parameter out of limits list values coincide with scripted telemetry data and associated limits flags.

Test Case 2 Parameter Out of Limit List - EOC - Back Orbit from GSFC DAAC

This test verifies the ability to produce parameter out of limits lists from spacecraft/instrument telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry - available at FOS on-line archive and at GSFC DAAC. User directives to set up FOS parameter out of limits report activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Parameter out of limits list values/flags coincide with scripted telemetry data. Data received from the GSFC DAAC is used for parameter out of limits list generation.

Test Case 3 Parameter Out of Limit List - EOC - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce parameter out of limits list based on data not available at the FOS on-line archive and GSFC DAAC.

Input: User directives to set up FOS parameter out of limits list generation activity.

Output: Report display at the EOC user stations; event message displays.

Expected Results: Event messages at EOC user interface event page stating selected parameter out of limits list generation activity incomplete due to "no data" condition.

Test Case 4 Parameter Out of Limit List - EOC - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during a parameter out of limit list generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS parameter out of limits list generation activity.

Output: Event message display.

Expected Results: Event message at EOC user interface event page stating incorrect setup file entry; parameter out of limits list generation process not begun until all setup file information is valid.

Test Case 5 Parameter Out of Limit List - IST - All Data Available

This test verifies the ability to produce parameter out of limits lists (i.e., parameter mnemonic name and associated limits flag and time tag) over a specified time period from spacecraft/instrument telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS parameter out of limits list generation activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Parameter out of limits list values coincide with scripted telemetry data and associated limits flags.

Test Case 6 Parameter Out of Limit List - IST - Back Orbit from GSFC DAAC

This test verifies the ability to produce parameter out of limits lists from spacecraft/instrument telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry - available at FOS on-line archive and at GSFC DAAC. User directives to set up FOS parameter out of limits report activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Parameter out of limits list values/flags coincide with scripted telemetry data. Data received from the GSFC DAAC is used for parameter out of limits list generation.

Test Case 7 Parameter Out of Limit List - IST - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce parameter out of limits list based on data not available at the FOS on-line archive and GSFC DAAC.

Input: User directives to set up FOS parameter out of limits list generation activity.

Output: Report display at the IST user stations; event message displays.

Expected Results: Event messages at IST user interface event page stating selected parameter out of limits list generation activity incomplete due to "no data" condition.

Test Case 8 Parameter Out of Limit List - IST - Incorrect Set-Up File Entry

This test verifies the ability to report on incorrect set up file entry conditions during a parameter out of limit list generation session.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS parameter out of limits list generation activity.

Output: Event message display.

Expected Results: Event message at IST user interface event page stating incorrect setup file entry; parameter out of limits list generation process not begun until all setup file information is valid.

5.1.28.2.5 Sequence 5 - Customized Statistics

This sequence verifies those requirements related to non-automatic (custom) statistical telemetry analysis report submittal, generation and display activity performed via the EOC and IST user stations' standing order manager function.

Test Case 1 Custom Stats Generation - EOC - All Data Available

This test verifies the ability to produce min, max, mean and standard deviation parameter values for orbital, hourly, daily, weekly, and monthly reports from telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics for specified stats (min, max, mean, standard deviation) for the time period requested and those statistic(s) calculations coincide with scripted telemetry data value statistic computations.

Test Case 2 Custom Stats Generation - EOC - Data Sets Sent from GSFC DAAC

This test verifies the ability to produce orbital, hourly, daily, weekly, and monthly reports from telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics (min, mean, max, standard deviation) for the time period requested and those statistic calculations coincide with scripted telemetry data value statistic(s) computations. Data received from the GSFC DAAC is used for statistic computations; data received from the GSFC DAAC is archived at the FOS on-line archive.

Test Case 3 Custom Stats Generation - EOC - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce reports based on data not available at the FOS on-line archive or GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation activity.

Output: Report display at the EOC user stations; event message displays.

Expected Results: Event messages at EOC user interface event page stating report generation activity incomplete due to no data condition.

Test Case 4 Custom Stats Generation - EOC - Data Flags

This test verifies the ability to produce orbital, hourly, daily, weekly, and monthly reports and provide min, mean, max, and standard deviation values from telemetry data containing specific telemetry flags (i.e., limits, delta limits, static, bad quality).

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the EOC user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics for the time period requested and those statistic calculations coincide with scripted telemetry data value computations. Computation of parameter statistics with any value "out of limit" results in "out of limit" result. Computation of parameter statistics with any value "out of delta limit" results in "out of delta limit" result. Computations of parameter statistics do not include any parameter sample(s) flagged as bad quality or static.

Test Case 5 Custom Stats Generation - EOC - Graphical Representation

This test verifies the ability to produce various Stats reports in graphical report format.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report status display at the EOC user stations; graph stats report printouts.

Expected Results: Graphical output of requested mnemonics identify calculated statistics for the time period requested and those statistic calculations coincide with scripted telemetry data value computations. Y axis min and max value ranges automatically calculated; max value higher than the highest value of any requested parameter; min value lower than the lowest value of any requested parameter. Times on graph grid coincide with time values specified for reporting period.

Test Case 6 Custom Stats Generation - EOC - Incorrect Set Up File

This test verifies the ability to report incorrect set up file input (i.e., incorrect mnemonics, incorrect start/stop times, incorrect data set name) upon statistical report generation request.

Input: User directives to set up FOS report generation activity.

Output: Event message display at EOC.

Expected Results: Event message stating incorrect input specification. Report request is not "submitted" until all report request input fields are valid.

Test Case 7 Custom Stats Generation - IST - All Data Available

This test verifies the ability to produce min, max, mean and standard deviation parameter values for orbital, hourly, daily, weekly, and monthly reports from telemetry data residing in the FOS on-line archive.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics for specified stats (min, max, mean, standard deviation) for the time period requested and those statistic(s) calculations coincide with scripted telemetry data value statistic computations.

Test Case 8 Custom Stats Generation - IST - Data Sets Sent from GSFC DAAC

This test verifies the ability to produce orbital, hourly, daily, weekly, and monthly reports from telemetry data residing in the FOS on-line archive and the GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics (min, mean, max, standard deviation) for the time period requested and those statistic calculations coincide with scripted telemetry

data value statistic(s) computations. Data received from the GSFC DAAC is used for statistic computations; data received from the GSFC DAAC is archived at the FOS on-line archive.

Test Case 9 Custom Stats Generation - IST - No Data Available

This test verifies the ability to report on "no data" condition when attempting to produce reports based on data not available at the FOS on-line archive or GSFC DAAC.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation activity.

Output: Report display at the IST user stations; event message displays.

Expected Results: Event messages at IST user interface event page stating report generation activity incomplete due to no data condition.

Test Case 10 Custom Stats Generation - IST - Data Flags

This test verifies the ability to produce orbital, hourly, daily, weekly, and monthly reports and provide min, mean, max, and standard deviation values from telemetry data containing specific telemetry flags (i.e., limits, delta limits, static, bad quality).

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report display at the IST user stations; report printouts.

Expected Results: Report output of requested mnemonics identify calculated statistics for the time period requested and those statistic calculations coincide with scripted telemetry data value computations. Computation of parameter statistics with any value "out of limit" results in "out of limit" result. Computation of parameter statistics with any value "out of delta limit" results in "out of delta limit" result. Computations of parameter statistics do not include any parameter sample(s) flagged as bad quality or static.

Test Case 11 Custom Stats Generation - IST - Graphical Representation

This test verifies the ability to produce various Stats reports in graphical report format.

Input: Scripted and archived spacecraft/instrument housekeeping and health & safety telemetry. User directives to set up FOS report generation and reporting activity.

Output: Report status display at the IST user stations; graph stats report printouts.

Expected Results: Graphical output of requested mnemonics identify calculated statistics for the time period requested and those statistic calculations coincide with scripted telemetry data value

computations. Y axis min and max value ranges automatically calculated; max value higher than the highest value of any requested parameter; min value lower than the lowest value of any requested parameter. Times on graph grid coincide with time values specified for reporting period.

Test Case 12 Custom Stats Generation - IST - Incorrect Set Up File

This test verifies the ability to report incorrect set up file input (i.e., incorrect mnemonics, incorrect start/stop times, incorrect data set name) upon statistical report generation request.

Input: User directives to set up FOS report generation activity.

Output: Event message display at IST.

Expected Results: Event message stating incorrect input specification. Report request is not "submitted" until all report request input fields are valid.

5.1.28.2.6 Sequence 6 - Standing Order Manager

This sequence of tests verifies those requirements relating to stat report standing order management functions, including multiple stat report processing, stat report management (i.e., deleting, halting, starting job queues), and requesting stats report generation status.

Test Case 1 Multiple Stats Report Requests

This test verifies the ability to perform multiple stats report requests via the FOS UIF's Standing Order Manager function.

Input: User directives to set up FOS report generation activity at both EOC and IST user stations. Previously generated report setup files at all user stations.

Output: Event message display at EOC/IST user stations. Displayed report queuing activity. Stats report printouts.

Expected Results: Event messages stating stats report queuing activity at all job requesting user stations. Following report job completion, report printable, viewable at requesting EOC/IST user stations; next stats report in queue is automatically generated and viewable/printable at requesting user station until report queue is empty.

Test Case 2 Standing Order Browser View

This test verifies the ability to view all standing orders in the FOS system, view standing order status (i.e., pending jobs, current job % complete) at EOC and IST user stations.

Input: User directives to set up standing order browse activity and report generation requests.

Output: Event message and standing order browse display at EOC and IST user stations. Displayed report queuing activity. Stats report printouts.

Expected Results: Event messages stating stats report queuing activity. Following report job completion, report printable, viewable at requesting EOC and IST user stations; next stats report in queue is automatically generated and viewable/printable until report queue is empty. Current job shows % complete.

Test Case 3 Standing Order Halt, Resume, Delete

This test verifies the ability to halt and resume the standing order queue, and delete entries from the queue by authorized EOC users.

Input: User directives to set up standing order browse, halt, resume, delete activity.

Output: Event messages and standing order browse display at EOC and IST user stations. Displayed report queuing activity.

Expected Results: Event messages stating stats report queuing activity. Halt, resume and delete commands allowed from authorized users only. Upon halt, all report activity halts until resume directive is issued. Any non-active report job, when deleted from the report queue, is no longer in the report queue directory.

Test Case 4 Saving Stats Results

This test verifies the ability to save stats report results for later printing/processing at EOC/IST user stations.

Input: User directives to save stats report files.

Output: Event message display at the EOC user stations.

Expected Results: All saved result files may be re-printed and/or re-displayed. Any non-saved stat report files are deleted following sign-off of the user station.

5.1.29 FOS Services Build (B14)

The FOS Services Build is an integrated build of the following system threads:

- FOS User Interface
- FOS Data Management
- FOS Resource Management
- FOS Planning & Scheduling
- FOS Command Management
- FOS Spacecraft Commanding

- FOS Telemetry
- FOS Analysis

5.1.29.1 Build Objectives

The objectives of the FOS Services Build are as follows:

- Provide FOS system access and security, system management and communication services in support of AM-1 flight operations.
- Provide the capability to process AM-1 spacecraft/instrument telemetry.
- Provide the capability to plan and schedule both spacecraft/instrument activity and flight operations ground resources needed to fulfill the AM-1 mission goals.
- Provide the capability to generate and transmit spacecraft/instrument commands and command loads.
- Provide the capability to store spacecraft/instrument telemetry and to provide historical analysis products from the stored telemetry.
- Provide the capability to perform AM-1 flight operation activity from a graphical user interface.
- Provide connectivity to FOS external interfaces including SDPS, FDF, NCC, EDOS, ECOM and the ASTER IP-ICC.

5.1.29.2 Build Test Description

Several EOC and IST user stations will configure to the AM-1 operational configuration, several EOC and IST user stations will configure to the AM-1 simulation/test/development configuration. External communications links will be established. AM-1 flight operations activities will then be performed including spacecraft/instrument commanding, scheduling, telemetry receipt and processing, and EOC/IST user station configuration.

Dependencies: (If Applicable)

In order to verify the objectives of the FOS Services Build, all eight FOS service threads are required.

Test Support Requirements

- Hardware:
  - EOC/IST User Stations
  - FOS Real-time Data Server
  - FOS Data Archive Server
  - FOS Operational LAN
  - AM-1 spacecraft/instrument real-time telemetry simulator
- Data:
  - AM-1 spacecraft/instrument, health & safety telemetry

– AM-1 spacecraft/instrument plans/schedules

– Historical telemetry residing in the FOS Data Archive

– AM-1 spacecraft/instrument command and command load data

5.1.29.2.1 Sequence 1 - FOS Operational

The following series of tests verify the ability to perform pre-contact, contact, and post-contact spacecraft/instrument flight operations activities simultaneously in support of the AM-1 spacecraft mission.

Test Case 1 AM-1 Flight Operation Support - Pre Contact

This test verifies that pre-contact flight operation activities taking place before spacecraft contact (i.e., planning and scheduling, command management, coordination of FDF planning aid transmission, schedule coordination with the NCC) may be performed simultaneously with AM-1 analysis support activity taking place at other EOC and IST user stations.

Input: User directives input at EOC/IST user stations to configure the FOS operational configuration including the ASTER IP-ICC external interface. Plans and schedules (i.e., BAPs, IALs, IADLs); user directives to coordinate the generation and transmission of plans and schedules.

Output: User interface status and event windows; status event messages. Spacecraft/instrument plan and schedule snaps, activity profile snaps, command and command load displays and dumps. Ground script displays and snaps. AM-1 spacecraft/instrument parameter graph snaps and displays.

Expected Results: AM-1 spacecraft/instrument plans and schedules, including the coordination of FDF and NCC information used for planning and scheduling aids is coordinated and performed by the user interface functions provided at the EOC and IST locations. IST input to plans and schedules is performed without impacting ongoing IST/EOC operational activity. AM-1 support activity performed via the EOC/IST user stations in no way affects performance or output of planning & scheduling and command management output.

Test Case 2 AM-1 Flight Operations Support - Spacecraft Contact

This test verifies flight operations activities performed in support of the AM-1 spacecraft and its complement of instruments during real-time contact periods. Operational activity verified includes the ability to receive and process various spacecraft/instrument telemetry types, perform spacecraft/instrument commanding and the coordination of the EDOS and ECOM external interfaces needed for contact support. Also verified is the ability to perform all spacecraft contact activity via the EOC and IST functional user interface capabilities.

Input: User directives input at EOC/IST user stations to configure the FOS operational configuration including EDOS, ECOM external interfaces. User directives at EOC and IST user stations to configure for telemetry and command operational and support functions.

Output: User interface status and event windows; status event messages, telemetry window displays/snaps, command uplink window displays/snaps. Event status messages at EOC/IST user stations.

Expected Results: AM-1 spacecraft/instrument telemetry processing is performed at both EOC and IST locations; commanding activity allowed in conjunction with other real-time, non-real time support activity. Spacecraft command activity allowed at MASTER controller only and in conjunction with telemetry processing; command output matches data base command definitions. Command and telemetry functionality is in no way affected by other support activity taking place at EOC and IST user stations.

Test Case 3 AM-1 Flight Operations Support - Post Contact

This test verifies flight operations activities performed in support of the AM-1 spacecraft and its complement of instruments during post-contact periods. Operational activity verified includes the ability to archive incoming spacecraft/instrument telemetry, update spacecraft/instrument plans and schedules, and provide historical analysis reports/graphs to FOS users.

Input: User directives input at EOC/IST user stations to configure for FOS analysis, planning & scheduling and support activity. User directives to coordinate the initialization of the SDPS interface.

Output: User interface status and event windows; status event messages, telemetry window displays/snaps, command uplink window displays/snaps, historical analysis reports. event status messages at EOC/IST user stations.

Expected Results: AM-1 spacecraft/instrument historical analysis processing is performed at both EOC and IST locations in conjunction with planning & scheduling and other support activity. Analysis processing and display mirrors requested data and correct value/flag information. Post contact operational activity performed at EOC/IST user stations in no way affects concurrent AM-1 support activity.

5.2 TRMM (Rel A) System Build

5.2.1 TRMM (Rel A) Build (B15)

The TRMM (Rel A) System Build is an integrated build of the following system threads and builds:

- System Integration Build 4
- TRMM Ingest Thread
- AM-1 Ingest Thread
- Two-Way V0 Interoperability Thread
- NOAA Interoperability Thread
- FOS Services Build

5.2.1.1 Build Objectives

The objectives of the TRMM (Rel A) Build are:

- Provide the capability to perform spacecraft activity scheduling, commanding, and telemetry monitoring operations.
- Provide system and network security.
- Provide the capability to perform various system functions concurrently. This will include (but is not limited to) user logon and registration, user query, ingesting of data, pre-processing of data, processing of ingested data, re-processing of data for distribution, and distribution of data.
- Provide the capability to monitor all on-going system activities (numerous functions being performed simultaneously), introduce a major fault condition, and recover from the fault condition. Recovery may require the re-prioritizing of on-going system activities.
- Provide the ability to extract data from the archive to support data/product distribution and for analytical purposes.

5.2.1.2 Build Test Description

Testing will demonstrate the ability to perform all spacecraft operations. Test activities will include generation of a spacecraft plan and command schedule, commanding of the spacecraft, receipt of spacecraft engineering and health and safety data, real-time monitoring of spacecraft engineering and health and safety data, archiving of engineering and spacecraft health and safety data, and receipt of science instrument data at the DPF.

Testing will demonstrate the ability to perform various activities concurrently on the ECS system. Test activities will include numerous user queries and data searches, new user logon and registration, ingesting of PDS data packets from the DPF, pre-processing of data, processing ingested data based on a planning schedule, re-processing of data in support of data distribution requests, and data distribution. User authorization and authentication to data requested will be verified prior to transfer of any data/products requested.

Testing will demonstrate the ability to detect and recover from fault conditions at various levels within the system. With each type of fault introduced, testing will demonstrate the ability to recover from the fault condition and verify that non-related system functions are not impacted by an extraneous fault condition (i.e., a processing fault does not cause system degradation at the remote access level).

Dependencies: (If Applicable)

In order to verify the objectives of the TRMM (Rel A) Build, the following interface and functional capabilities are required:

Simulated DPF facility/environment.

Simulated External Interfaces for DAACs, ADCs, EPDSs, ODCs PIs, TLs, TMs, and SCFs.

Test Support Requirements

- Hardware:
  - TBS
- Software:
- Data:
  -

5.2.1.2.1 Sequence 1 - Spacecraft Scheduling, Commanding and Monitoring

The following test verifies the capability to plan and schedule both spacecraft and ground resource activities. Testing will include the generation of spacecraft command and instrument loads, uplinking of the command and instrument loads, real-time receipt and monitoring of spacecraft engineering and health and safety data, monitoring of engineering and spacecraft health and safety data, archiving of health and safety data, and real-time receipt of science instrument data to the DPF.

Test Case 1 FOS Scheduling, Commanding, Monitoring and Data Receipt

This test verifies the capability to plan and schedule spacecraft activities, uplinking of command and instrument loads, receipt/monitoring of spacecraft health and safety data, and the downlinking of data. The ability to archive incoming heath and safety data is demonstrated.

Input: User directives input at EOC/IST user stations. Plans and schedules (i.e., BAPs, IALs, IADLs); user directives to coordinate the generation and transmission of plans and schedules.

User directives input at EOC/IST user stations to configure the FOS operational configuration including EDOS, ECOM external interfaces. User directives at EOC and IST user stations to configure for telemetry and command operational and support functions.

User directives input at EOC/IST user stations to configure for FOS analysis, planning & scheduling and support activity. User directives to coordinate the initialization of the SDPS interface.

Output: User interface status and event windows; status event messages. Spacecraft/instrument plan and schedule snaps, activity profile snaps, command and command load displays and dumps. Ground script displays and snaps. AM-1 spacecraft/instrument parameter graph snaps and displays.

Telemetry window displays/snaps, command uplink window displays/snaps. Event status messages at EOC/IST user stations.

Expected Results: AM-1 spacecraft/instrument plans and schedules, including the coordination of FDF and NCC information used for planning and scheduling aids are coordinated and performed by the user interface functions provided at the EOC and IST locations. IST input to plans and schedules is performed without impacting ongoing IST/EOC operational activity. AM-1 support activity performed via the EOC/IST user stations in no way affects performance or output of planning & scheduling and command management output.

AM-1 spacecraft/instrument telemetry processing is performed at both EOC and IST locations; commanding activity allowed in conjunction with other real-time, non-real time support activity. Spacecraft command activity allowed at COMMAND ACTIVITY controller only and in conjunction with telemetry processing; command output matches data base command definitions. Command and telemetry functionality is in no way affected by other support activity taking place at EOC and IST user stations.

AM-1 spacecraft/instrument historical analysis processing is performed at both EOC and IST locations in conjunction with planning & scheduling and other support activity. Analysis processing and display mirrors requested data and correct value/flag information. Post contact operational activity performed at EOC/IST user stations in no way affects concurrent AM-1 support activity.

5.2.1.2.2 Sequence 2 - Concurrent System Activity

The following test verifies the ability to perform concurrent activities on the ECS system. Data queries, data searches, and requests for data and products will be performed while data is being ingested, pre-processing of data as well as additional data processing is ongoing. Re-processing of data in support distribution requests is performed. Two-way V0 Interoperability is demonstrated. The ability to monitor all transactions between Non-ECS and ECS elements is verified.

Test Case 1 System Activity

This test verifies the ability to perform concurrent activities on the ECS System.

Input: Existing processing plan, scheduled allocation of resources based on the plan. Plan includes ingesting raw data from the DPF, ingesting of L0-L4 data sets, special product data, metadata, ancillary data, calibration data, correlative data, algorithms, and documentation from a DAAC, an ADC, a SCF, and other science community users (TLs, TMs PIs).

On-going data processing based on the plan, multiple users performing logon and registration activity, previously registered users performing data searches/queries at the data dictionary, data server and LIM levels, multiple users 'browsing' guide data, multiple users 'browsing' the advertising service and installing the service to the users workbench, generation and submission of data distribution requests (some requiring re-processing of data prior to distribution). Distribution requests will support both a 'push' and a 'pull' of the data.

On-going communications between ECS and V0 networks. Updates are made to existing metadata listings and data access services and advertised to all clients. Multiple users performing a browse, searching, and ordering data products from V0 and ECS.

Users requesting status of outstanding distribution requests and pending processing requests.

Output: Allocation/De-allocation of resources based on processing plan and processing completion, generation of data availability notice upon data processing completion/storage in the archive.

Ingest log contains an entry for each data type ingested, readiness notification if data ingest is coming from the DPF, initiation of data validation, completion of data validation, and notification to the PGS to perform standard product production (if required).

Security logs contain entries for generation of access/authorization level for each new user registered and verification of each users access/authorization to data requested prior to the data being distributed.

Event logs, history logs contain entries for user logon, user searches and queries performed at each level, users 'browsing' the guide data, and users 'browsing' and installing advertising services.

Data requiring re-processing prior to distribution is worked into the plan. Originator of the request is notified of expected processing completion time. 'Pull' request data is located and originator of the request is notified of the location of the data. 'Push' request

data is gathered and distributed to the originator of the request based on information contained in the distribution request. Distribution of the data is monitored. Data distribution log contains entries for each distribution request fulfilled.

Status messages generated to users providing status of outstanding or pending distribution/processing requests.

Expected Results: Resources are properly allocated and de-allocated to support product generation. The guide and directory are updated as data is stored to the data server upon processing completion, a data availability notice is generated and issued when data is written to the archive, LIM data is updated based on the data availability notice.

Ingested data is validated and written to temporary storage to support product generation. PGS receives notification of data requiring processing. Data is removed from temporary storage upon processing completion.

User logon and registration results in update to the access control list and generation of a user profile.

User access/authorization to data or product requested is verified prior to data distribution.

User receives response to a submitted status request for the disposition of an outstanding distribution or processing request.

5.2.1.2.3 Sequence 3 - Supporting System Faults

The following test verifies the ability to re-prioritize or reschedule system activities as a result of a system malfunction (fault). This test is executed several times, introducing a system malfunction (fault) at different levels of the system to verify system wide impact and recovery.

Test Case 1 System Malfunction

This test verifies the ability to re-prioritize system activities as a result of a system malfunction while other concurrent activities are on-going on the ECS System.

Input: Existing processing plan, scheduled allocation of resources based on the plan. Plan includes ingesting raw data from the DPF, ingesting of L0-L4 data sets, special product data, metadata, ancillary data, calibration data, correlative data, algorithms, and documentation from a DAAC, an ADC, a SCF, and other science community users (TLs, TMs PIs).

On-going data processing based on the plan, multiple users performing logon and registration activity, previously registered users performing data searches/queries at the data dictionary, data server and LIM levels, multiple users 'browsing' guide data,

multiple users 'browsing' the advertising service and installing the service to the users workbench, generation and submission of data distribution requests (some requiring re-processing of data prior to distribution). Distribution requests will support both a 'push' and a 'pull' of the data.

On-going communications between ECS and V0 networks. Updates are made to existing metadata listings and data access services and advertised to all clients. Multiple users performing a browse, searching, and ordering data products from V0 and ECS.

Introduce a system fault at various system levels (i.e., processing level, remote access level, storage level).

Output: HP OpenView detects fault and site status color changes to reflect the fault condition.

If the fault is located at the processing level, a processing failure occurs. Resource availability marked 'bad'. Processing plan updated with processing re-routed to another resource. Changes in the plan reflect new expected completion times for all remaining items in the queue. Users notified of changes to expected completion times of all data/product requests pending. .

Data requiring pre-processing prior to distribution is re-worked into the plan. Originator of the request is notified of new expected processing completion time. 'Pull' request data is located and originator of the request is notified of the location of the data. 'Push' request data is gathered and distributed to the originator of the request based on information contained in the distribution request. Distribution of the data is monitored. Data distribution log contains entries for each distribution request fulfilled.

If the fault is not at the processing level, processing should not be impacted

If the fault is located at the ingest level, current ingest processing terminates. Ingest log contains an entry for the failure of the ingest. If the fault is not at the ingest level, normal ingesting continues. Ingest log contains an entry for each data type ingested, readiness notification if data ingest is coming from the DPF, initiation of data validation, completion of data validation, and notification to the PGS to perform standard product production (if required).

Security logs contain entries for generation of access/authorization level for each new user registered and verification of each users access/authorization to data requested prior to the data being distributed.

If the fault is located at the access level, remote access is terminated, event and history logs contain an entry for the fault condition. If the fault is not at the access level, event logs, history logs contain entries for user logon, user searches and queries performed at each level, users 'browsing' the guide data, and users 'browsing' and installing advertising services.

Expected Results: HP OpenView can be used to aid in locating and isolating the fault condition.

If a processing fault, processing plan is updated as a result of loss of resource allocation. Resources are properly allocated and de-allocated to support product generation based on new plan. Users are notified of new 'expected' processing completion times. The guide and directory are updated as data is stored to the data server upon processing completion, a data availability notice is generated and issued when data is written to the archive, LIM data is updated based on the data availability notice.

If fault is at the ingest level, ingest is terminated. Originator is notified to re-transmit the data upon system restoration. Ingested data is validated and written to temporary storage to support product generation. PGS receives notification of data requiring processing. Data is removed from temporary storage upon processing completion. Loss of the processing resource has no impact on data ingestation but may impact time scheduled processing time.

If fault is at the remote access level, access to the system is terminated. If a new user is performing the registration process at the time of the fault, all data must be re-entered at the next logon attempt. The access control list and user profiles are not updated as a result of the fault.

If the fault is between ECS and V0 during an update of metadata listings, the update is terminated without any updates being posted to the data base. Upon correction of the fault condition, a re-transmission request for the metadata is issued.

If the fault is not at the remote access level, user logon and registration results in update to the access control list and generation of a user profile. Loss of the processing resource has no impact on users performing logon and registration.

User access/authorization to data or product requested is verified prior to data distribution. A fault condition should never result in security degradation.

5.2.1.2.4 Sequence 4 - SMC Directives and Data Availability Schedules

The following series of tests verify the ability to receive SMC scheduling directives to support training activities, maintenance activities, and emergency situations. Testing also verifies the ability to receive data availability schedules from EDOS, ADCs, other DADs and the SDPF.

Test Case 1 System Resource Scheduling - SMC Directives

This test verifies the ability to generate a plan which requires allocation of a resource to support preventive maintenance or training activities for a specified period of time as a result of a SMC directive. The resource is being utilized by the system prior to receipt of the SMC directive.

Input: SMC directive requiring a resource be allocated to support a training or preventive maintenance activity for a specified time period. Resource currently allocated to support upcoming or current activities based on the existing plan.

Output: Plan re-generated based on resource not being available from time period 'x to y'. Resource removed from the plan for the specified time period. Resource allocation in the plan noted again after time period 'y' has expired. Resource log contains an entry for the removal of the device and the change of status.

Expected Results: Current processing on the allocated resource completes prior to its removal to support the training or preventative maintenance activity. Resource status reflects 'unavailable' for time period 'x to y'.

Test Case 2 System Resource Scheduling - SMC Emergency Directives

This test verifies the ability to generate a plan which requires allocation of a resource to support an emergency situation as a result of a SMC directive. The resource is being utilized by the system prior to receipt of the SMC directive.

Input: SMC directive requiring a resource be allocated to support an emergency situation, anticipated duration of time required not known. Resource currently allocated to support upcoming or current activities based on the existing plan.

Output: Plan re-generated based on resource not being available for an indefinite period of time. Resource removed from the plan and marked as 'unavailable'. Resource log contains an entry for the removal of the device and the change of status.

Expected Results: Current processing on the allocated resource terminated prior to its completion. New plan allocates required resources to complete processing that was in progress prior to the emergency plan re-generation. If processing based on a user request, user provided with new 'expected' processing completion time.

Test Case 3 Receiving Data Availability Schedules

This test verifies the ability to receive data availability schedules. Receipt of schedules from EDOS, and ADC, DADs and SDPF will be demonstrated.

Input: Data availability schedule from external interface under test (EDOS, ADC, DAD, and SDPF).

Output: Notification of receipt of data availability schedule from external interface under test. Alarm posted to operator terminal for receipt of schedule. Event log contains an entry for the receipt of the data availability schedule.

Expected Results: Schedule received, alarm posted, entry made to the event log.

Test Case 4 SMC Security Directives

This test verifies the ability to receive security directives from the SMC.

Input: Security directive from the SMC.

Output: Notification of receipt of security directive from the SMC. Alarm posted to operator terminal for receipt of directive. Event log contains an entry for the receipt of the security directive.

Expected Results: Directive received, alarm posted, entry made to the event log.

5.2.1.2.5 Sequence 5 - Supporting Data Analysis/Algorithm Development

The following series of tests verify the ability to provide instrument science teams access to stored data to support data analysis and algorithm development.

Test Case 1 Testing an Algorithm

This test verifies the capability for science team members to access stored data to use in performing data processing in support of generation of a new algorithm or updating an existing algorithm.

Input: Stored data in the archive. New algorithm under development or modified existing algorithm. Sufficient resources to perform offline processing with the algorithm under test. Results package.

Output: Status messages, processing messages, product data, results package.

Expected Results: Successful access to the data stored in the archive. Any error conditions encountered during processing with algorithm under test logged in the processing log. Termination of processing logged. Successful processing resulted in generation of a results package.

5.2.1.2.6 Sequence 6 - Receipt of Various Types of Data - System Notifications

The following series of tests verify the ability to receive L0-L4 data, L0-L4 equivalent data sets, special products, metadata, ancillary data, calibration data, correlative data, documents, and

algorithms from other DAACs, ADCs, EPDSs, ODCs, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Ingestation of the data is performed and written to temporary storage. The interface between ingest and data server management is verified with the proper generation of a data availability notice by ingest to the data server controller. Successful permanent storage should occur. Upon successful storage, the interface with the LIM will be verified as a result of the successful generation of a data availability notice and the LIM information being updated.

Test Case 1 QA'd Product Metadata

This test verifies the capability to receive updated metadata for products that have been QA'd.

Input: Updated metadata for a QA'd product. Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of the new metadata, new metadata written to temporary storage, metadata verified, and readiness notification issued to data server management for data archival. Data archival completed and advertisement issued for new data availability. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: Updated metadata successfully ingested and written to temporary storage. Metadata verified. Ingest processor notifies data server management of data availability for storage. Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 2 QA'd Product Metadata - Verification Failure

This test verifies the capability to receive updated metadata, verify its content, and issue appropriate failure/status messages as a result of verification failure.

Input: Updated metadata for a QA'd product that contains invalid parameters. Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of the new metadata, new metadata written to temporary storage, metadata verified, and error conditions detected. Ingest generates failure notification message to originator. Data removed from temporary storage. Ingest log contains an entry for ingest initiation, data failure, and generation of notification to the originator.

Expected Results: Metadata ingested, data verification fails, and originator notified of failure condition. Data deleted from temporary storage. Corresponding logs contain entries for events transpired.

Test Case 3 Receipt of L0-L4 Data

This test verifies the capability to receive L0 through L4 data. Testing verifies the ability to receive data from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: L0 through L4 data. External interface with DAAC, ADC, EPDS, an ODC, TL, TM, PI, Co, and a SCF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of data from the external source, data written to temporary storage, and readiness notification issued to data server management for data archival. Data availability notice issued to product processing for processing of the data. Upon successful processing, product processing issues notification to data server management for product archival readiness. Processing log contains an entry for the successful processing of the data. Data archival completed and advertisement issued for new data availability. Event log contains an entry for advertisement of data to the LIM. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: New L0-L4 data successfully ingested and written to temporary storage. Data verified. Ingest processor notifies data server management of data availability for storage and product processing for data processing. Data processing completes and notifies data server controller of product availability for data storage. Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 4 Receipt of Ancillary Data - Pre-processing Required

This test verifies the capability to receive ancillary data requiring pre-processing. Testing verifies the ability to receive data from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: Ancillary data requiring pre-processing. External interface with DAAC, ADC, EPDS, an ODC, a TL, a TM, a PI, a Co, and a SCF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of ancillary data from the external source, data written to temporary storage, and readiness notification issued to data server management for data archival and within ingest to perform pre-processing. Upon successful processing, ingest issues notification to data server management for data archival readiness. Ingest log contains an entry for the successful pre-processing of the data. Data archival completed and advertisement issued for new data

availability. Event log contains an entry for advertisement of data to the LIM. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: Ancillary data successfully ingested and written to temporary storage. Data verified. Ingest processor notifies data server management of data availability for storage and pre-processing Processing completed and data server controller notified of data availability for storage. Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 5 Receipt of Ancillary Data - No Pre-processing Required

This test verifies the capability to receive ancillary data. Testing verifies the ability to receive data from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: Ancillary data. External interface with DAAC, ADC, EPDS, an ODC, a TM., a TL, a PI, a Co, and a SCF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of ancillary data from the external source, data written to temporary storage, and readiness notification issued to data server management for data archival Ingest log contains an entry for the successful ingest of the data. Data archival completed and advertisement issued for new data availability. Event log contains an entry for advertisement of data to the LIM. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: Ancillary data successfully ingested and written to temporary storage. Data verified. Ingest processor notifies data server management of data availability for storage. Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 6 Receipt of Calibration Data

This test verifies the capability to receive calibration data. Testing verifies the ability to receive data from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: Calibration data. External interface with DAAC, ADC, EPDS, an ODC, a TM, a TL, a PI, a Co, and a SCF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of calibration data from the external source, data written to temporary storage, and readiness notification issued to data server management for data archival Data archival completed and advertisement issued for new data availability. Event log contains an entry for advertisement of data to the LIM. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: Calibration data successfully ingested and written to temporary storage. Data verified. Ingest processor notifies data server management of data availability for storage Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 7 Receipt of Correlative Data

This test verifies the capability to receive correlative data. Testing verifies the ability to receive data from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: Correlative data. External interface with DAAC, ADC, EPDS, an ODC, a TM, a TL, a PI, a Co, and a SCF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of correlative data from the external source, data written to temporary storage, and readiness notification issued to data server management for data archival Data archival completed and advertisement issued for new data availability. Event log contains an entry for advertisement of data to the LIM. LIM data is updated. Users are able to achieve access to the new data.

Expected Results: Correlative data successfully ingested and written to temporary storage. Data verified. Ingest processor notifies data server management of data availability for storage Data successfully written to the archive and an advertisement is issued to update the LIM. A client/user is able to access the data. Directory and inventory reflects addition of new data.

Test Case 8 Receipt of Algorithms

This test verifies the capability to receive an algorithm package. Testing verifies the ability to receive algorithm packages from a DAAC, an ADC, an EPDS, an ODC, ECS science community (TLs, TMs, PIs, Cos), and a SCF.

Input: Algorithm package. External interface with DAAC, ADC, EPDS, an ODC, a TM, a TL, a PI, a Co, and a SCF. (interface may be

simulated or stubbed). Ingestation ready to receive data notification.

Output: Ingest log contains an entry for the successful ingestation of algorithm package from the external source, algorithm written to temporary storage, and readiness notification issued to data server management for algorithm archival Algorithm archival completed. Ingest issues notification to PGS for receipt of new algorithm requiring verification. PGS verification completed. Notification issued by PGS of successful algorithm verification. Data server issues advertisement of addition of new algorithm to the LIM. LIM data updated to advertise availability of the algorithm.

Expected Results: Algorithm package successfully ingested and written to temporary storage. Ingest processor notifies data server management of data availability for storage Data successfully written to the archive and PGS notified of need to perform algorithm verification. Algorithm verified, PGS notifies data server controller of successful verification, data server issues advertisement to update the LIM. A client/user is able to access the algorithm. Directory and inventory reflects addition of new algorithm.

5.2.1.2.7 Sequence 7 - Receipt of FDF Data - AM-1 Early Interface

The following series of tests verify the ability to receive orbit, attitude, and metadata from the FDF.

Test Case 1 Receipt of Orbit Data

This test verifies the capability to receive orbit data. Testing verifies the ability to receive data from the FDF. Testing is performed to support early interface testing of the AM-1 platform.

Input: Orbit data. External interface with FDF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of orbit data from the external source, data written to temporary storage. Ingest log contains an entry for the initiation of data ingest, success of ingest, and data staging.

Expected Results: Orbit data written to the staging area.

Test Case 2 Receipt of Attitude Data

This test verifies the capability to receive attitude data. Testing verifies the ability to receive data from the FDF. Testing is performed to support early interface testing of the AM-1 platform.

Input: Attitude data. External interface with FDF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Data ingest log contains an entry for the successful ingestation of attitude data from the external source, data written to temporary storage. Ingest log contains an entry for the initiation of data ingest, success of ingest, and data staging.

Expected Results: Attitude data written to the staging area.

Test Case 3 Receipt of Metadata

This test verifies the capability to receive metadata data. Testing verifies the ability to receive data from the FDF. Testing is performed to support early interface testing of the AM-1 platform.

Input: FDF metadata. External interface with FDF. (interface may be simulated or stubbed). Ingestation ready to receive data notification.

Output: Ingest log contains an entry for the successful ingestation of metadata from the external source, data written to temporary storage. Ingest log contains an entry for the initiation of metadata ingest, success of ingest, and metadata staging.

Expected Results: Metadata written to the staging area.

5.2.1.2.8 Sequence 8 - Generating Metadata Items

The following series of tests verify the ability to generate metadata items to support unique granule id for L0 data, date and time of data storage, physical data location, data validation status, and unique format identifiers.

Test Case 1 Unique Granule ID

This test verifies the capability to generate a unique granule id for data written to the archive.

Input: L0 data. Ingestation ready to receive data notification.

Output: L0 data ingested, written to temporary storage, data verified, and data server management notified of data ready for archival. Archival of data to unique granule id performed. Data server management generates notification of data availability to LIM. LIM data updated with new data to granule id level. Users/clients able to access data based on granule id. Ingest log contains an entry for the successful ingest, data verification, and notification to data server manager. Data server log contains an entry for the addition of new data and advertisement to the LIM.

Expected Results: L0 data ingested, verified, and written to permanent archive. Inventory and directory updated with L0 data to the granule level. Advertisement of granule id level data issued to the LIM. LIM data updated. Users/clients access the data.

Test Case 2 Assigning Storage Criteria Data to Metadata

This test verifies the capability to generate metadata with an associated date and time stamp, physical data location, data consistency status, and other unique identifiers for data stored in the archive.

Input: New data or updates to existing archived data. Ingestation ready to receive data notification.

Output: Data ingested, written to temporary storage, data verified, and data server management notified of data ready for archival. Archival of data performed. Date and time stamp, physical archival location, consistency check status, and unique format identifiers associated with the data is saved with the data in the archive. Data server management generates notification of data availability to LIM. LIM data updated with new data. Users/clients able to access data. Ingest log contains an entry for the successful ingest, data verification, and notification to data server manager. Data server log contains an entry for the addition of new data and advertisement to the LIM.

Expected Results: Data ingested, verified, and written to permanent archive. Inventory and directory updated with data. Advertisement of data issued to the LIM. LIM data updated. Users/clients able to access the data.

5.2.1.2.9 Sequence 9 - Gathering Statistical Data

The following series of tests verify the ability to gather statistical data and generate reports as required.

Test Case 1 Storage Media Statistics

This test verifies the capability to collect statistical information on the raw and corrected bit error rates for data ingested and written to the archive.

Input: Canned set of raw data with known acceptable bit error conditions. Ingestation ready to receive data notification.

Output: Data ingested, written to temporary storage, data verified, bit error conditions noted, and data server management notified of data ready for archival. Archival of data performed. Bit error conditions information saved with data set. Data server management generates notification of data availability to LIM. LIM data updated with new data. Users/clients able to access data. User receives related bit error rate data with the metadata

associated with the data set. Ingest log contains an entry for the successful ingest, data verification, bit error conditions noted for the data set, and notification to data server manager. Data server log contains an entry for the addition of new data and advertisement to the LIM.

Expected Results: Data ingested, verified, and written to permanent archive. Inventory and directory updated with data. Advertisement of data issued to the LIM. LIM data updated. Users/clients able to access the data. User receives bit error condition data in the metadata associated with the data set.

5.2.1.2.10 Sequence 10 - Distributing Various Types of Info/Data

The following series of tests verify the ability to send L0-L4 data, special products, metadata, ancillary data, calibration data, correlative data, documents, and algorithms to PGS, IMS, other DAACs, ADCs, users/clients, ECS science community (TLs, TMs PIs Cos), IPs, and SCFs. Testing will verify the ability to support multiple data transfers simultaneously and the ability to prioritize distribution requests.

Test Case 1 L0-L4 Data

This test verifies the capability to send L0-L4 data to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: L0-L4 data resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 2 Special Product Data

This test verifies the capability to send special product data to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), IPs, and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Special product data resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 3 Metadata

This test verifies the capability to send metadata to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Metadata resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 4 Ancillary Data

This test verifies the capability to send ancillary data to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Ancillary data resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 5 Calibration Data

This test verifies the capability to send calibration data to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include

requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Calibration data resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 6 Correlative Data

This test verifies the capability to send correlative data to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Correlative data resident in the archive. Distribution requests for data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Data located in the archive, authorization to receive data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data set.

Expected Results: Data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 7 Documents

This test verifies the capability to send documents to IMS, other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Documents resident in the archive. Distribution requests for document information from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Document located in the archive, authorization to receive document requested verified, and document transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each document

Expected Results: Document distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 8 Algorithms

This test verifies the capability to send algorithms to other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Algorithms and supporting algorithm data (test data, historical info, etc) resident in the archive. Distribution requests for algorithm and all supporting documentation (test data, test results, algorithm history info, etc) from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Algorithm and supporting documentation located in the archive, authorization to receive algorithm and related documentation requested verified, and algorithm package transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each algorithm package.

Expected Results: Algorithm package distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 9 Data Granules

This test verifies the capability to send data at the granule level to other DAACs, ADCs, users, ECS science community (TLs, TMs, PIs, Cos), and SCFs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Granule data resident in the archive. Distribution requests for granule data from all interfaces under test (DAAC, ADC, etc). Request originated with an assigned priority, with at least two requests having the same priority.

Output: Granule data located in the archive, authorization to receive granule level data requested verified, and data transferred to the requesters based on priority. Distribution log contains an entry for each distribution request and transfer of each data granule.

Expected Results: Granule data distributed to the originator based on priority of request. Priority conflicts resolved by the system.

Test Case 10 Spacecraft Data

This test verifies the capability to send spacecraft related information (command loads, planning and scheduling data, etc) to IPs. Testing will include requests from authorized and non-authorized users to verify system security. Multiple requests with the same and varying priorities will be issued concurrently.

Input: Request from an IP for spacecraft related data.

Output: Request received, authorization to receive requested data verified, data distributed to request originator. Distribution log contains an entry for the distribution of the requested data.

Expected Results: User authorization to receive data verified. Data distributed to requester.

5.2.1.2.11 Sequence 11 - System Analysis and Inspections

The following series of tests verify system/user models developed will support system growth/expansion without requiring major architecture changes, meet security requirements for system startup and shutdown, meet required performance standards, meet reliability predictions and maintainability predictions.

Test Case 1 Growth, Expansion, and Testing

This test verifies that user models demonstrate the ability for growth and expansion without requiring a major system redesign. Report findings should also demonstrate the ability to perform testing on the system during all phases of its development.

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: System will support growth and expansion without requiring a major system redesign. System testing can be performed during all phases of system development.

Test Case 2 System Shutdown/Startup

This test verifies that user models demonstrate the ability to provide security safeguards and both scheduled and un-scheduled system shutdowns and startups.

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: Security safeguards for system startup and shutdown have been accounted for.

Test Case 3 Performance Standards

This test verifies that user models demonstrate the ability to adhere to GSFC 420-05-03 performance assurance requirements for EOSDIS.

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: Report findings demonstrate the ability to adhere to required performance standards.

Test Case 4 Reliability Predictions

This test verifies that user models demonstrate the ability to adhere to reliability predictions based on MIL-HDBK-217F (Reliability Prediction of Electronic Equipment).

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: Report findings demonstrate the ability to adhere to required reliability predictions.

Test Case 5 Maintainability Predictions

This test verifies that user models demonstrate the ability to adhere to maintainability predictions based on MIL-HDBK-472 (Maintainability Prediction), MIL-STD-470A (Maintainability Program) and GSFC 420-05-03 (Performance Assurance Requirements).

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: Report findings demonstrate the ability to adhere to required maintainability predictions.

Test Case 6 Data Flow

This test verifies that user models demonstrate the ability to adhere to data flow requirements at the element level.

Input: Reports generated by user model working group.

Output: Review of report findings.

Expected Results: Report findings demonstrate the ability to adhere to required element data flows.

5.2.1.2.12 Sequence 12 - System/Data Recovery

The following series of tests verify the capability to recover from system errors. The ability to recover lost data as a result of a catastrophic error is demonstrated.

Test Case 1 Data Lost - Reproduce from L0

This test verifies the capaiblity to reproduce product data from L0 data as a result of data loss.

Input: Removal of data product from the archive. Existing L0 data to support data removed from the archive. Resource allocation to reprocess data and produce product data.

Output: Advertisement issued to reflect loss of data from the archive. LIM data updated, removing data entry for lost data. User unable to obtain access to data lost. Processing of L0 data completed. New product data generated to recover lost data. Processing subsystem

issues notification to data server management of processing completion. Product data written to permanent archive. Advertisement issued for recovered data. LIM receives notification. LIM data updated. Users able to access reproduced product data.

Expected Results: When data is lost from the archive (removed for testing purposes), an advertisement is issued to support updates (removal of data at the LIM level). Data is reprocessed from L0 and product reproduced. New product is stored to the archive and advertisement is issued to update the LIM data. Directory and inventory updated with new data. User is able to access product data.

5.2.1.2.13 Sequence 13 - Accessing ODCs and ADCs

The following series of tests verify the capability to perform level 2 and level 3 catalog interoperability on data contained at an ODC or ADC.

Test Case 1 ODC Data Access

This test verifies the capability to obtain level 2 and level 3 access to data at an ODC.

Input: Known data set at an ODC. Access to ODC system.

Output: User granted access to data set located at ODC. Event log contains an entry for the connectivity to the ODC system and receipt of data requested from ODC.

Expected Results: Data access granted. Data transferred from ODC to local workstation.

Test Case 2 ADC Data Access

This test verifies the capability to obtain level 2 and level 3 access to data at an ADC.

Input: Known data set at an ADC. Access to ADC system.

Output: User granted access to data set located at ADC. Event log contains an entry for the connectivity to the ADC system and receipt of data requested from ADC.

Expected Results: Data access granted. Data transferred from ADC to local workstation.

5.3 TRMM (Rel A) Interface Testing

The external ECS interfaces for TRMM (Rel A), as described in the following paragraphs will be tested. Both external and internal interfaces are tested as a part of the build/thread test methodology. Each thread and/or build describes the interfaces that must be in place to support the testing of that build and/or thread.

5.3.1 TRMM Mission and Ground System

The TRMM platform relies on ECS to support its mission. TSDIS will process the level 0 data from TRMM instruments (PR, TMI and VIRS). ECS will provide data archive for this data. Additionally, ECS will provide the production facilities for processing and storage of higher level CERES and LIS TRMM data. The TRMM release will support end-to-end ECS operations to include the generation of DAAC specific (GSFC, LaRC, and MSFC) processing plans, incorporation of schedule adjudications, resource management, data processing, data ingest, data storage, and data distribution to users/clients. The ECS will support user/client generation of data and product distribution requests, status requests, on-demand processing requests, and distribution/processing cancellations. New clients/users shall be able to logon to the system complete a registration form and obtain the ECS Scientist Workbench to their local workstation. User profiles and access privileges shall be established/created to prevent unauthorized disclosure of data or product information. All system activities to include software usage, resource usage, user/client logons, and data transfers shall be monitored and recorded for analytical reporting and system fault monitoring.

5.3.2 Landsat-7 Early Interface

Early interface testing of the Landsat 7 mission is provided in the TRMM release. ECS is responsible for ingesting and archiving Landsat 7 L0R and product data for the Landsat 7 mission. Early interface testing will verify the ability to ingest and stage Landsat 7 L0R data, metadata, and product data from the IGS and as many as 16 international partners. Post TRMM release, ECS is responsible for the permanent archiving of Landsat 7 L0R data, metadata, and product data as well as distributing the data based on user/client requests/queries.

5.3.3 Early AM-1 Interface

The flight Operations Segment (FOS) will support early interface testing with the spacecraft, instruments, EDOS, and ECOM. Testing will verify the ability to perform spacecraft command, control, and monitoring. This includes planning and scheduling of spacecraft activities, generation and uplinking of spacecraft command loads, real-time receipt of scientific and engineering data, monitoring of spacecraft health and safety data, and archival of instrument data for analytical and report generation purposes.

5.4 TRMM (Rel A) Regression Testing

Testing of each thread will take place independently. As each thread is integrated into a build, that build will be regression tested to verify that no functionality or capability was lost/broken as a result of the integration. In addition, testing of new capabilities that have resulted from the build integration will be performed and verified.

Where a build focuses on interfacing with other areas of the system, regression testing will focus on testing the interface capabilities/functionalities as a result of the integration rather than the underlying thread functionality. Regression testing to ensure no interface capability/functionality was lost/broken will be performed and testing of new interface capabilities will be performed and verified.

5.5 TRMM (Rel A) Site Integration Testing

A month before the completion of the TRMM (Rel A) integration tests (or before Consent to Ship Review), a preliminary version of the ECS software system will be installed and tested for functionality at an appropriate DAAC. It is assumed that all required hardware will be installed and operational at the designated DAAC in order to conduct the test.

Abbreviations and Acronyms

APIs	Application Program Interfaces
ASF	Alaska SAR Facility
ASTER	Advanced Spaceborne Thermal Emission and Reflection Radiometer (formerly ITIR)
ATC	Absolute Time Command
ATMP	Acceptance Test Management Plan
AVHRR	Advanced Very High Resolution Radiometer
BB	Bulletin Board
CCSDS	Consultative Committee for Space Data Systems
CD-ROM	Compact disk -- read only memory
CDRL	Contract Data Requirements List
CERES	Clouds and Earth's Radiant Energy System
CIESIN	Consortium for International Earth Science Information Network
CM	Configuration Management
CMS	Command Management System (for UARS)
COTS	Commercial Off the Shelf
CSMS	Communications and System Management Segment
CTV	Compatibility Test Van
DAAC	Data Analysis and Archive Center
DADS	Data Archive Distribution System
DAS	Direct Access System (AM-1)
DAT	Digital Audio Tape
DIF	Data Interface Facility (EDOS); directory interchange format
DID	Data Item Description
DR	Discrepancy Report
ECOM	EOS Communications
ECS	EOSDIS Core System
EDC	EROS Data Center
EDF	Engineering Development Facility
EDOS	EOS Data and Operations System

EOC	Earth Observation Center (Japan); EOS Operations Center (ECS)
EOS	Earth Observing System
EOSDIS	Earth Observing System Data Information System
EPs	Evaluation Packages
ESN	EOSDIS Science Network
ETR	Element Test Review
EU	Engineering Units
F&PRS	Functional and Performance Requirements Specification
FDF	Flight Dynamics Facility
FOS	Flight Operations Segment
FTM	File Transfer Management
GSFC	Goddard Space Flight Center
HDF	Hierarchical Data Format
HMI	Human machine interface
I&T	Integration & Test
I&TT	Integration & Test Team
IAL	Instrument Activity List
IADL	Instrument Activity Deviation List
IATO	Independent Acceptance Test Organization
ICC	Instrument Control Center
ICDs	Interface Control Documents
IDB	Instrument Data Base
IGSE	Instrument Ground Support Equipment
IMS	Information Management System
IRDs	Interface Requirements Documents
IRP	Instrument Resource Profile
IST	Instrument Support Terminal
IV&V	Integration, Verification and Validation
JPL	Jet Propulsion Laboratory
Kbps	Kilobit/kilobyte per second
LAN	Local Area Network
LaRC	Langley Research Center
LTSP	Long-Term Science Plan

LTIP	Long-Term Instrument Plan
MISR	Multi-angle Imaging Spectro-Radiometer
MODIS	Moderate Resolution Imaging Spectrometer
MOPITT	Measurements of Pollution in the Troposphere
MR	Malfunction/Failure Report
MSFC	Marshall Space Flight Center
NASA	National Aeronautics and Space Administration
NCC	Network Control Center
NOAA	National Oceanic and Atmospheric Administration
O&M	Operations & Maintenance
ODB	Operations Data Base
ORB	Object Request Broker
ORNL	Oak Ridge National Laboratory
P&S	Planning and Scheduling
PAIP	Performance Assurance Implementation Plan
PDR	Preliminary Design Review
PGS	Product Generation System
PR	Precipitation Radar (TRMM)
PRS	Preliminary Resource Schedule
PVP	Performance Verification Plan
RIT	Real-Time Integrated Testbed
RTC	Relative Time Commands
RTS	Relative Time Sequence
S/E	Systems Engineering
SCF	Science Computing Facilities
SDPF	Sensor Data Processing Facility (GSFC)
SDPS	Science Data Processing Segment
SDR	System Design Review
SDS	System Design Specification
SDVF	Software Development & Validation Facility
SeaWifs	Sea-Viewing Wide Field-of-View Sensor
SEDAAC	Socio-Economic Data And Applications Center (CIESIN)
SEIO	System Engineering & Integration Office

SEP	System Engineering Plan
SIP	System Implementation Plan
SI&T	Systems Integration and Test
SITP	System Integration Test Plan
SMC	System Management Center
SRR	System Requirements Review
SSR	Solid State Recorder
SSRP	Spacecraft Subsystem Resource Profile
TBD	To Be Defined/Determined
TBR	To be replaced/resolved/reviewed
TBS	To Be Specified
TDRS	Tracking and Data Relay Satellite
TDRSS	Tracking and Data Relay Satellite System
TMI	TRMM Microwave Image
TSDIS	TRMM Science Data and Information System
VIRS	Visible Infrared Scanner (TRMM)
VP	Verification Plan
VS	Verification Specification

Glossary

acceptance testing	Verification that is conducted to determine whether a release satisfies its acceptance criteria and that provides the Government with information for determining whether the release should be accepted. Acceptance testing also applies to toolkits, science algorithm integration, and unit-level verification of COTS products.
activity	A specified amount of scheduled work that has a defined start date, takes a specific amount of time to complete, and comprises definable tasks.
affiliated data center (ADC)	A facility not funded by NASA that processes, archives, and distributes Earth science data useful for global change research, with which a working agreement has been negotiated by the EOS program. The agreement provides for the establishment of the degree of connectivity and interoperability between EOSDIS and the ADC needed to meet the specific data access requirements involved in a manner consistent and compatible with EOSDIS services. Such data-related services to be provided to EOSDIS by the ADC can vary considerably for each specific case.
algorithm	Software delivered to the SDPS by a science investigator (PI, TL, or II) to be used as the primary tool in the generation of science products. The term includes executable code, source code, job control scripts, as well as documentation.
analysis	Technical or mathematical evaluation based on calculation, interpolation, or other analytical methods. Analysis involves the processing of accumulated data obtained from other verification methods.
attitude data	Data that represent spacecraft orientation and onboard pointing information. Attitude data includes: • Attitude sensor data used to determine the pointing of the spacecraft axes, calibration and alignment data, Euler angles or quaternions, rates and biases, and associated parameters. • Attitude generated onboard in quaternion or Euler angle form. • Refined and routine production data related to the accuracy or knowledge of the attitude.

auxiliary data	Auxiliary data can be any data set which enhances the processing or utilization of satellite remote sensing instrument data. The auxiliary data is not captured by the same data collection process as the instrument data. Auxiliary data sets can include data collected by any platform or process, preferably in georeferenced digital format (CEOS).
availability	A measure of the degree to which an item is in an operable and committable state at the start of a "mission" (a requirement to perform its function) when the "mission" is called for an unknown (random) time. (Mathematically, operational availability is defined as the mean time between failures divided by the sum of the mean time between failures and the mean down time [before restoration of function].
baseline	Identification and control of the configuration of software (i.e., selected software work products and their descriptions) at given points in time.
baseline, *configuration management specific*	A configuration identification document or a set of such documents formally designated by the Government at a specific time during a configuration item's life cycle. Baselines, plus approved changes from those baselines, constitute the current approved configuration identification.
baseline activity profile	A schedule of activities for a target week corresponding to normal instrument operations constructed by integrating long term plans (i.e.,, LTSP, LTIP, and long term spacecraft operations plan).
browse data product	1. Subsets of a larger data set, other than the directory and guide, generated for the purpose of allowing rapid interrogation (i.e.,, browse) of the larger data set by a potential user. For example, the browse product for an image data set with multiple spectral bands and moderate spatial resolution might be an image in two spectral channels, at a degraded spatial resolution. The form of browse data is generally unique for each type of data set and depends on the nature of the data and the criteria used for data selection within the relevant scientific disciplines.
build	An assemblage of threads to produce a gradual buildup of system capabilities.
build, *configuration management specific*	An assemblage of threads to produce a gradual buildup of system capabilities. Builds are combined with other builds and threads to produce higher level guilds.

calibration	The collection of data required to perform calibration of the instrument science data, instrument engineering data, and the spacecraft engineering data. It includes pre-flight calibration measurements, in-flight calibrator measurements, calibration equation coefficients derived from calibration software routines, and ground truth data that are to be used in the data calibration processing routine.
Capabilities and Requirements Review (CRR)	Assessment of EOSDIS annual Project level capabilities and requirements. CRR is conducted to determine if ECS is meeting its objectives and provides evolutionary direction for new or modified requirements. Current information is provided on how ECS supports the EOS mission.
catalog system	An implementation of a directory, plus a guide and/or inventories, integrated with user support mechanisms that provide data access and answers to inquiries. Capabilities may include browsing, data searches, and placing and taking orders. The system is a specific implementation of a catalog service.
command	Instruction for action to be carried out by a space-based instrument or spacecraft.
commercial off-the-shelf (COTS)	“Commercial off-the-shelf” means a product, such as an item, material, software, component, subsystem, or system, sold or traded to the general public in the course of normal business operations at prices based on established catalog or market prices.
component	The next lower functional subdivision below "subsystem" in the ECS functional hierarchy.
comprehensive and incremental scheduling	Two modes of scheduling. Comprehensive scheduling is the automatic scheduling of a full set of events. Incremental scheduling is interactive scheduling of selected events. For example, the initial generation of a schedule might use comprehensive scheduling, while the addition of a single event with the desire to avoid perturbing previously scheduled events might use incremental scheduling.
computer software component (CSC)	A distinct part of a computer software configuration item (CSCI). CSCs may be further decomposed into other CSCs and computer software units (CSUs).
computer software configuration item (CSCI)	A configuration item comprised of computer software components (CSCs) and computer software units (CSUs).
computer software unit (CSU)	An element specified in the design of a computer software component (CSC) that is separately testable. A package of work allocated to an individual member of the software development team (CSUs do not span personnel boundaries).

configuration	The functional and physical characteristics of hardware, firmware, software or a combination thereof as set forth in technical document and achieved in a product.
configuration control	The systematic proposal, justification, evaluation, coordination, approval or disapproval of proposed changes, and the implementation of all approved changes in the configuration of a configuration item after formal establishment of its baseline.
configuration item (CI)	An aggregation of hardware, firmware, software or any of its discrete portions, which satisfies an end use function and is designated for configuration management.
configuration item, *configuration management specific*	An aggregation of hardware, firmware, software or any of its discrete portions, which satisfies an end use function and is designated for configuration control. CIs are those items whose performance parameters and physical characteristics must be separately defined (specified) and controlled to provide management insight needed to achieve the overall end use function and performance.
consent to ship review (CSR)	Review to determine the readiness of a release for transition to sites for integration testing.
correlative data	Scientific data from other sources used in the interpretation or validation of instrument data products, e.g., ground truth data and/or data products of other instruments. These data are not utilized for processing instrument data.
critical design review (CDR)	A detailed review of the element/segment-level design, including such details as program design language (PDL) for key software modules, and element interfaces associated with a release.
critical path	A path on a network which has the greatest negative slack and is the longest path in time through the network.
data acquisition request (DAR)	A request for future data acquisition by an instrument(s) that the user constructs and submits through the IMS.
data archive and distribution system	Included in each DAAC [Distributed Active Archive Center], and responsible for archiving and distribution of EOS data and information.
data availability schedule	Data availability schedule is a schedule indicating the times at which specific data sets will be available from remote DADS, EDOS, the IPs, the ADCs and ODCs for ingestion by the collocated DADS. The schedules are received directly by the PGS.
data center	A facility storing, maintaining, and making available data sets for expected use in ongoing and/or future activities. Data centers provide selection and replication of data and needed documentation and, often, the generation of user tailored data products.

data product

A collection (1 or more) of parameters packaged with associated ancillary and labeling data. Uniformly processed and formatted. Typically uniform temporal and spatial resolution. (Often the collection of data distributed by a data center or subsetted by a data center for distribution.) There are two types of data products:

- Standard–A data product produced at a DAAC by a community consensus algorithm. Typically produced for a wide community. May be produced routinely or on-demand. If produced routinely, typically produced over most or all of the available independent variable space. If produced on-demand, produced only on request from users for particular research needs typically over a limited range of independent variable space.
- Special–A data product produced at a science computing facility by a research status algorithm. May migrate to a community consensus algorithm at a later point. If adequate community interest, may be archived and distributed by a DAAC.

data product levels

• Raw data--Data in their original packets, as received from the observer, unprocessed by EDOS.

• Level 0--Raw instrument data at original resolution, time ordered, with duplicate packets removed.

• Level 1A--Reconstructed unprocessed instrument data at full resolution, time referenced, and annotated with ancillary information, including radiometric and geometric calibration coefficients and georeferencing parameters (i.e., platform ephemeris) computed and appended, but not applied to Level 0 data.

• Level 1B--Radiometrically corrected and geolocated Level 1A data that have been processed to sensor units.

• Level 2--Derived geophysical parameters at the same resolution and location as the Level 1 data.

• Level 3--Geophysical parameters that have been spatially and/or temporally re-sampled (i.e.,, derived from Level 1 or Level 2 data).

• Level 4--Model output and/or results of lower level data that are not directly derived by the instruments.

Data levels 1 through 4 as defined in the EOS Data Panel Report. Consistent with the CODMAC and ESADS definitions.

data quality request

Data quality request is a request issued by the PGS to a scientist at an SCF to perform QA of a particular product before future processing or distribution. A time window is applied to the request in keeping with the production schedule.

data rate profile	Instrument's and subsystem's data rate requirements (e.g., recorder data rate) to be included in the instrument or subsystem resource profiles.
dataset	A logically meaningful grouping or collection of similar or related data.
dataset documentation	Information describing the characteristics of a data set and its component granules, including format, source instrumentation, calibration, processing, algorithms, etc.
demonstration	Observation of the functional operation of the verification item in a controlled environment to yield qualitative results without the use of elaborate instrumentation, procedure, or special test equipment.
design interface	The interaction and relationship of logistics with the system engineering process to ensure that system element influences the definition and design of system elements so as to reduce life cycle costs.
detailed activity schedules	The schedule for a spacecraft and instruments which covers up to a10 day period and is generated/updated daily based on the instrument activity listing for each of the instruments on the respective spacecraft. For a spacecraft and instrument schedule the spacecraft subsystem activity specifications needed for routine spacecraft maintenance and/or for supporting instruments activities are incorporated in the detailed activity schedule.
deviation	A written request to depart temporarily (for a specific period of time or number of units) from the authorized baseline requirements (i.e.,, a temporary or limited waiver).
directory	A collection of uniform descriptions that summarize the contents of a large number of data sets. It provides information suitable for making an initial determination of the existence and contents of each data set. Each directory entry contains brief data set information (e.g., type of data, data set name, time and location bounds).

Distributed Active Archive Center (DAAC)	An EOSDIS facility which generates, archives, and distributes EOS Standard Products and related information for the duration of the EOS mission. An EOSDIS DAAC is managed by an institution such as a NASA field center or a university, per agreement with NASA. Each DAAC contains functional elements for processing data (the PGS), for archiving and disseminating data (the DADS), and for user services and information management (elements of the IMS). DAAC Sites: EDC – EROS Data Center ASF – Alaska SAR Facility LaRC – Langley Research Center GSFC – Goddard Space Flight Center JPL – Jet Propulsion Laboratory MSFC – Marshall Space Flight Center NSIDC – National Snow and Ice Data Center
DAAC-unique	Functions and capabilities provided by the DAAC beyond those provided by the Core System. The functions will be integrated with ECS via APIs for other similar mechanisms. Examples of DAAC-unique functions include visualization, specialized interfaces, and dataset-unique functionality.
ECS evolutionary development	The process for delivering and evolving ECS functionality through the used of multiple development tracks and delivery mechanisms. Use of development tracks and delivery mechanisms are tailored to the goals of the particular portion of the system of the system, with an overall goal of providing relatively stable portions of the system in comparison to portions which are rapidly adapting to the system's environment.
ECS project	ECS provides single-point access for a worldwide science community, simultaneous mission management for multiple instruments, and regular production of validated science products using community-supplied algorithms.
ECS-supported	A hardware or software component that conforms to an ESDIS approved set of standards and has been fully tested by the ECS contractor.

EDOS production data set	Data sets generated by EDOS using raw instrument or spacecraft packets with space-to-ground transmission artifacts removed, in time order, with duplicate data removed, and with quality/ accounting (Q/A) metadata appended. Time span, number of packets, or number of orbits encompassed in a single data set are specified by the recipient of the data. These data sets are equivalent to level zero data formatted with Q/A metadata. For EOS, the data sets are composed of: • instrument science packets • instrument engineering packets, • spacecraft housekeeping packets, or • onboard ancillary packets with quality and accounting information from each individual packet and the data set itself and with essential formatting information for unambiguous identification and subsequent processing.
EDOS quick look production data sets	Data sets generated by EDOS using raw instrument or spacecraft packets from a single TDRSS acquisition session and made available for delivery to a user within 1 hour of receipt of the last packet in the session. Transmission artifacts are removed, but time ordering and duplicate packet removal is limited to packets received during the TDRSS contact period.
element	The next lower functional subdivision below "segment" within the ECS functional hierarchy.
element test review (ETR)	Determines if development level testing (for each release) has successfully been completed.
evaluation package	An evaluation package is a delivery mechanism for incrementally developed components and selected prototypes. The objectives of evaluation packages are to increase user involvement in system evolution and rapid evaluation and to facilitate rapid incorporation of user feedback into the incremental development process.
formal development track	A development process distinguished by complete tree of requirements documentation, formal reviews at major milestones in the development cycle and a single waterfall of phases leading to a formal release. The single waterfall has a long time frame relative to the incremental development track and prototypes.
formal qualification testing	A process that allows the contracting agency to determine whether a configuration item complies with the allocated requirements for that item.

formal release	A formal release (Release) is a system-wide update to the ECS, delivered and tested as a part of the EOSDIS. ECS Releases will represent the ECS portion of EOSDIS Versions. Formal releases are part of the formal development track.
function	The action or actions which an item is designed to perform.
functional baseline	The initial baseline established at the system requirements review (SRR) and refined at the system design review (SDR).
functional configuration audit (FCA)	The formal examination of functional characteristics of a CI, prior to acceptance, to verify that the item has achieved the performance specified in its functional or allocated configuration identification.
granule	The smallest aggregation of data that is independently managed (i.e.,, described, inventoried, retrievable). Granules may be managed as logical granules and/or physical granules.
granule location	The name of the product where this granule is located.
guide	A detailed description of a number of data sets and related entities, containing information suitable for making a determination of the nature of each data set and its potential usefulness for a specific application.
hardware	That combination of subcontracted, commercial-off-the-shelf (COTS), and government furnished equipment (GFE) (e.g., cables and computing machines that are the platforms for software)
hardware configuration item (HWCI)	A configuration item comprised of hardware components.
incremental design review	Review conducted to evaluate segment designs associated with a release.
incremental development track	A development process distinguished by multiple iterations of requirements, design, and implementation with frequent user evaluations via demonstrations. Documentation and reviews are streamlined. Documentation of non-mission critical is created after development has completed. Each increment is developed with the potential of being integrated into the formal track for a release. The incremental development track has a cycle time between the formal development and prototypes.
incremental scheduling	See "comprehensive and incremental scheduling.

independent verification and validation (IV&V)	Verification and validation performed by a contractor or government agency that is not responsible for developing the product or performing the activity being evaluated. IV&V is an activity that is conducted separately from the software development activities governed by the ECS contract.
in operations (or operational)	An ECS capability is in operations if some of the following are true: • It is accessible by non-ECS personnel (no matter how restricted the group may be) • Some or all of it resides outside of the EDF • It is used to support outside agencies/projects (including testing of interfaces with such) Operations does not necessarily imply responsibility of the formal M&O organization. For example, the early EPs are operated by a combination of the ECS developers and the site liaisons.
inspection	The visual, manual examination of the verification item and comparison to the applicable requirement or other compliance documentation, such as engineering drawings.
instrument	A hardware system that collects scientific or operational data. Hardware-integrated collection of one or more sensors contributing data of one type to an investigation. An integrated collection of hardware containing one or more sensors and associated controls designed to produce data on/in an observational environment.
instrument data	Data specifically associated with the instrument, either because they were generated by the instrument or included in data packets identified with that instrument. These data consist of instrument science and engineering data, and possible ancillary data.
instrument activity deviation list	An instrument's activity deviations from an existing instrument activity list, used by the EOC for developing the detailed activity schedule.
instrument activity list	An instrument's list of activities that nominally covers seven days, used by the EOC for developing the detailed activity schedule.
instrument microprocessor memory loads	Storage of data into the contents of the memory of an instrument's microprocessor, if applicable. These loads could include microprocessor-stored tables, microprocessor-stored commands, or updates to microprocessor software.

instrument resource deviation list	An instrument's anticipated resource deviations from an existing resource profile, used by the EOC for establishing TDRSS contact times and building the preliminary resource schedule.
instrument resource profile	Anticipated resource needs for an instrument over a target week, used by the EOC for establishing TDRSS contact times and building the preliminary resource schedule.
instrument science data	Data produced by the science sensor(s) of an instrument, usually constituting the mission of that instrument.
integration	The orderly progression of combining lower level software and/or hardware items to form higher level items with broader capability.
interface(s)	The functional and physical characteristics required to exist at a common boundary.
interim release	The delivery of system capability resulting from early efforts on the formal track development to the customer for testing of EOS functionality prior to an operational version.
interoperability	Refers to the capability of the user interface software of one data set directory or catalog to interact with the user interface at another data set directory or catalog. Three levels of Catalog Interoperability are recognized: • Level 1 - Simple network interconnectivity among systems. • Level 2 - Catalog systems can exchange limited search and user information. • Level 3 - Catalog systems exchange standard search protocols. This provides "virtual" similarity between different systems.
inventory	A uniform set of descriptions of granules from one or more data sets with information required to select and obtain a subset of those granules. Granule descriptions typically include temporal and spatial coverage, status indicators, and physical storage information. An inventory may describe physical granules, logical granules, or both, including a mapping between them if they are not identical. Note that the inventory is not the granules themselves, but rather the descriptive data for each of them, specifically used by both system and user to locate those desired.
long-term instrument plan (LTIP)	The plan generated by the instrument representative to the spacecraft's IWG with instrument-specific information to complement the LTSP. It is generated or updated approximately every six months and covers a period of up to approximately 5 years.

long-term science plan (LTSP)	The plan generated by the spacecraft's IWG containing guidelines, policy, and priorities for its spacecraft and instruments. The LTSP is generated or updated approximately every six months and covers a period of up to approximately 5 years.
long term spacecraft operations plan	Outlines anticipated spacecraft subsystem operations and maintenance, along with forecasted orbit maneuvers from the FDF, spanning a period of several months.
maintainability	The measure of the ability of an item to be retained in or restored to a specified condition when maintenance is performed by personnel having specified skill levels, using prescribed procedures and resources, at each prescribed level of maintenance and repair. (The probability that maintenance, both corrective and preventive, can be performed in a specified amount of time using a specified set of prescribed procedures and resources expressed as MTTR). Maintainability is the function of design.
maintenance	The process of planning and executing life cycle maintenance concepts and requirements necessary to ensure sustained operation of system elements.
maintenance downtime	The total elapsed time required to repair and restore a system to a full operational status and/or retain a system in that condition.
mean time between failure (MTBF)	The reliability result of the reciprocal of a failure rate that predicts the average number of hours that an item, assembly or piece part will operate within specific design parameters. (MTBF=1/(l) failure rate; (l) failure rate = # of failures/operating time.
mean time between maintenance (MTBM)	The average time between all maintenance including both corrective and preventive maintenance.
mean time to repair (MTTR)	The mean time required to perform corrective maintenance to restore a system/equipment to operate within design parameters.
metadata	Information about data sets which is provided to the ECS by the data supplier or the generating algorithm and which provides a description of the content, format, and utility of the data set. Metadata may be used to select data for a particular scientific investigation.
network	A flow diagram depicting the time phased sequence and interrelationship of events and activities that must be accomplished to achieve project objectives.
nonconformance	The failure of a unit or product to conform to specified requirements.
off-line	Access to information by mail, telephone, facsimile, or other non-direct interface.

on-line	Access to information by direct interface to an information data base via electronic networking.
operational data	Data created by an operational instrument (i.e.,, NOAA AMRIR).
orbit data	Data that represent spacecraft locations. Orbit (or ephemeris) data include: Geodetic latitude, longitude and height above an adopted reference ellipsoid (or distance from the center of mass of the Earth); a corresponding statement about the accuracy of the position and the corresponding time of the position (including the time system); some accuracy requirements may be hundreds of meters while other may be a few centimeters.
organizational breakdown	A functionally oriented pyramid-like structure indicating Structure organizational relationships and used as the framework for the assignment of responsibility.
parameter	The output generated by applying predetermined transformation algorithms to previously existing products and ancillary data, using specified calibration coefficients, to represent a specific geophysical parameter. Included are level 2-4 products.
physical configuration audit	The formal examination of the "as-built" configuration of a configuration item against its technical documentation to establish the CI's initial product configuration identification.
planning package	A future segment of work within a cost account that is not yet broken down into work packages. A planning package has a firm budget, estimated start and complete dates, and description of work.
platform	The EOS spacecraft and its subsystems without the instruments.
playback data	Data that have been stored on-board the spacecraft for delayed transmission to the ground.
Preliminary Design Review (PDR)	PDR is held for each ECS Segment. The PDR addresses the design of the segment-level capabilities and element interfaces through all ECS releases. The PDR also addresses prototyping results and how the results of both Contractor and Government prototyping efforts, studies, and user experience with EOSDIS Version 0 have been incorporated into the ECS design for each respective Segment.
preliminary resource schedule	An initial integrated spacecraft schedule, derived from instrument and subsystem resource needs, that includes the NCC TDRSS contact times and nominally spans seven days.
process	A logical sequence of tasks by which a job is accomplished.

product baseline	The baseline which establishes the "as-built" configuration for system-level integration and testing (I&T) and independent acceptance testing. This baseline is validated by functional and physical configuration audits, and reviewed and approved by the Goddard Space Flight Center (GSFC) as part of release readiness review.
product generation executive (PGE)	Product generation executive (PGE) is a set of one or more compiled binary executables and/or command language scripts; it is the smallest schedulable unit for PGS processing.
product generation executable	Product generation executable is an obsolete term. See product generation executive.
product order	Product order is either a request for the generation of a specific product with an associated time window, a priority processing request, a reprocessing request, or a standing order for a product to be generated on a regular basis with a rough timeline, or changes to standing orders. Product orders are received by the PGS from the IMS.
prototyping	The construction of a solution of a design or implementation problem, the feasibility of which needs to be determined as early as possible in order to arrive at a critical decision.
quality assurance	A subset of the total performance assurance activities generally focused on conformance to standards and plans.
quick-look data	1. Data received during one TDRSS contact period which have been processed to Level 0 (to the extent possible for data from a single contact). These are data that have been identified to EDOS as requiring priority processing.
quick-look product	produced at a PGS by applying science algorithms to quick-look data.

raw data	Data in their original packets, as received from the spacecraft and instruments, unprocessed by EDOS. • Level 0 – Raw instrument data at original resolution, time ordered, with duplicate packets removed. • Level 1A – Level 0 data, which may have been reformatted or transformed reversibly, located to a coordinate system, and packaged with needed ancillary and engineering data. • Level 1B – Radiometrically corrected and calibrated data in physical units at full instrument resolution as acquired. • Level 2 – Retrieved environmental variables (e.g., ocean wave height, soil moisture, ice concentration) at the same location and similar resolution as the Level 1 source data. • Level 3 – Data or retrieved environmental variables that have been spatially and/or temporally resampled (i.e.,, derived from Level 1 or Level 2 data products). Such resampling may include averaging and compositing. • Level 4 – Model output and/or variables derived from lower level data which are not directly measured by the instruments. For example, new variables based upon a time series of Level 2 or Level 3 data.
real-time data	Data that are acquired and transmitted immediately to the ground (as opposed to playback data). Delay is limited to the actual time required to transmit the data.
reconfiguration	A change in operational hardware, software, data bases or procedures brought about by a change in a system's objectives.
release initiation review (RIR)	An internal review conducted at the start of the development phase of a release to revisit the requirements and issues associated with that particular release.
release readiness review (RRR)	Conducted at the ECS system level for a GSFC project review team upon completion of release acceptance testing. The IATO leads the RRR to determine, with the GATT and the COTR, if the release is ready to be delivered, installed, and incorporated into the operational system.
regression testing	Re-test to insure the continued correct functioning usually after the replacement or addition of functionality.
requirement	A statement to which the developed system must comply. Varieties of requirements: levels 2, 3, 4; performance, functional, design, interface.

SCC-stored commands and tables	Commands and tables which are stored in the memory of the central onboard computer on the spacecraft. The execution of these commands or the result of loading these operational tables occurs sometime following their storage. The term “core-stored” applies only to the location where the items are stored on the spacecraft and instruments; core-stored commands or tables could be associated with the spacecraft or any of the instruments.
schedules	Schedules represent the current sequence of tasks to be executed along with approximate execution times as generated by the PGS scheduler. Copies of these schedules, which are updated frequently, are made available to the IMS, the SMC, and the DADS.
science computing facility	A facility supplied by the EOS program to an EOS team leader, team member, or principal investigator (instrument or interdisciplinary) for the following purposes: developing and maintaining the algorithms and software used to generate standard data products; quality control of standard data products; in-flight instrument calibration and data set validation; scientific analysis, modeling, and research' generation of special data products; and use as an interface to the investigator's institutional facility.
segment	One of the three functional subdivisions of the ECS, i.e.,, FOS, SDPS, and CSMS.
simulated data	simulated data - same as test data
software	A combination of associated computer instructions and computer data definitions required to enable the computer hardware to perform computational, data manipulation, and control functions (to include parameters and procedures associated with software products).
software development file	A repository for a collection of material pertinent to the development or support of software. Contents typically include (either direct or by reference) design considerations and constraints, design documentation and data, schedule and status information, test requirements, test cases, test procedures, and test results.
software development library (SDL)	A generic term which describes a controlled collection of software, documentation, and associated tools and procedures used to simplify the development and subsequent support of software. An SDL provides storage of and controlled access to software in both human readable and machine readable form. Also, it may contain management data pertinent to the software development project.
spacecraft subsystems activity list	A spacecraft subsystem's list of activities that nominally covers seven days, used by the EOC for developing the detailed activity schedule.

spacecraft subsystems resource profile	Anticipated resource needs for a spacecraft subsystem over a target week, used by the EOC for establishing TDRSS contact times and building the preliminary resource schedule.
special data products	Data products which are considered part of a research investigation and are produced for a limited region or time period, or data products which are not accepted as standard products.
specification	A document intended primarily for describing the essential technical requirements for items, material or services, including the procedures for determining whether or not the requirements have been met.
standard products	(a) Data products generated as part of a research investigation, of wide research utility, accepted by the IWG and the EOS Program Office, routinely produced, and in general spatially and/or temporally extensive. Standard Level 1 products will be generated for all EOS instruments; standard Level 2 products will be generated for most EOS instruments. (b) All data products which have been accepted for production at a PGS, including (1) above as well as prototype products.
statement of work	A description of the end objectives, scope, and constraints of a unique and separately identifiable portion of the work required to satisfy contract requirements.
status	Status is information regarding schedules, hardware and software configuration, exception conditions, or processing performance. This information is exchanged with the DADS, and is provided to the SMC. The SMC may also receive information regarding schedule conflicts that have not been resolved with the IMS.
subsampling	Standard subsampling involves extraction of a multi-dimensional rectangular array of pixels from a single data granule, where regularly-spaced, non-consecutive pixels are extracted from each array dimension. For each dimension, the size of the pixel array is characterized by the starting pixel location, the number of pixels to extract, and the pixel-spacing between extracted pixels.
subsetting	Standard subsetting involves extraction of a multi-dimensional rectangular array of pixels from a single data granule, where consecutive pixels are extracted from each array dimension. For each dimension, the size of the pixel array is characterized by the starting pixel location and the number of pixels to extract.
subsystem	A combination of sets, groups, etc., which performs an operational function~~al~~ within a system and is a major division of a system.

support equipment	All equipment required to support the operation and maintenance of a material system. This includes associated multi-use end items, ground handling and maintenance equipment, tools, calibration equipment, communications resources, test equipment and automatic test equipment with diagnostics software for both on-and-off equipment maintenance. It also includes the acquisition of logistics support for the support and test equipment itself.
system	A composite of equipment, skills, and techniques capable of performing or supporting an operational role (or both). A complete system includes all equipment, related facilities, material, software, services and personnel required for its operation and support to the degree that it can be considered a self-sufficient item in its intended operational environment.
System Design Review (SDR)	The SDR addresses the top-level ECS design. The SDR includes the definition and high-level design of ECS segments and elements, the interfaces between these and the interfaces between these and external systems, facilities, users, operators, etc.
System Requirements Review (SRR)	The SRR encompasses a complete review of the ECS specification and the EOS/EOSDIS Requirements (Level 2) that drive the specification, it promotes a common understanding between the Project and the Contractor of the capabilities that ECS must provide.
target of opportunity (TOO)	A TOO is a science event or phenomenon that cannot be fully predicted in advance, thus requiring timely system response or high-priority processing.
temporary file	Temporary file is a file which may exists for the duration of a single PGE, or may exist for some indeterminate time beyond the termination of the PGE which created it.
test	A procedure or action taken to determine under real or simulated conditions the capabilities, limitations, characteristics, effectiveness, reliability or suitability of a material, device, system or method.
testing	An element of inspection and generally denotes the determination by technical means of the properties or elements of supplies, or components thereof, including functional operation, and involves the application of scientific principles and procedures.
test data	Test data is any data set designed or specially selected to aid in the algorithm integration and test process, to help test the operation of algorithms under development by simulating realistic input.
test products	Test products are science products generated by new or updated algorithms during the integration and test period. Test products are delivered to scientists at an SCF.

test readiness review (TRR)	Conducted by the project for each release at the segment and element levels to review the plans for the integration and verification of the subsystems with the elements and the elements with their segments.
thread	A set of components (software, hardware, and data) and operational procedures that implement a function or set of functions.
toolkits	Some user toolkits developed by the ECS Contractor will be packaged and delivered on a schedule independent of ECS releases to facilitate science data processing software development and other development activities occurring in parallel with the ECS.
unit	An assembly of any combination of parts, subassemblies and assemblies mounted together which are normally capable of independent operation in a variety of situations.
user	Any person accessing the EOSDIS. • Authorized users are users who have viable EOSDIS accounts, and who may therefore make EOSDIS data requests. These users may be affiliated or unaffiliated. Affiliated users are those who are sponsored by one of the parties to the Earth Observations-International Coordination Working Group (EO-ICWG) data policy. Each party is responsible for ensuring that all its affiliated users comply with the EO-ICWG data policy. Use of data by affiliated users is classified in one of three categories, defined in the EO-ICWG data policy: + Research Use: A study or an investigation in which the user affirms (1) the aim is to establish facts or principles; (2) the data will not be sold or reproduced or provided to anyone not covered by this or another valid affirmation; (3) the results of the research will be submitted for publication in the scientific literature; and (4) detailed results of the research will be provided to the sponsoring spacecraft operator as agreed between the researcher and the sponsoring spacecraft operator. In the context of EOSDIS , this means that NASA-affiliated users must make available to the research community their detailed results, including data, algorithms, and models at the time their research is accepted for publication, and that the data may be copied and shared among other researchers provided that either they are covered by a research agreement or the researcher who obtained the data from EOSDIS is willing to take responsibility for their compliance with the agreement. Data for affiliated users and for research and applications use will be made available at no more than the marginal cost of production and distribution.

+ Environmental Monitoring and Operational Use: Includes data use by those government agencies affiliated with the parties which conduct environmental monitoring and/or operational observations for the public good, and can include larger agencies to which the parties belong (i.e.,, the World Meteorological Organization); or national agencies, or their designates, involved in other operational forecasting activities which are conducted for the public good (i.e.,, weather, sea state, sea ice, agriculture, hydrology, etc.). Environmental Monitoring and Operational Use of Data constitutes any use of data to carry out a mandate of environmental observation and prediction as part of an agency's responsibilities to provide for the general welfare. Such use may include the routine downlink or direct broadcast of enhanced and unenhanced data in near-real time within the operational community. Data for Environmental Monitoring and Operational use shall be provided in real or near-real time without fee, and shall be available through international EOS archives for non-real time users for no more than the marginal cost of reproduction and distribution consistent with the access terms for each instrument category.

+ Other users: Those persons requesting data for scientific, operational, applications, or commercial use, who are not directly represented by an EO-ICWG member, and who agree to the stipulations on data access and use as set by the EO-ICWG and the EOS program.

validation — The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements.

variance analysis — Identification of variation from a planned baseline and analysis to determine its scope, cause, impact, and corrective action.

verification — The process of evaluating the products of a given development activity to determine correctness and consistency with respect to the products and standards provided as input to that activity.

version — Versions are the culmination of a series of ECS releases, in conjunction with incorporation of SCF-developed science data processing software and unique site capabilities.

waiver, engineering — A written authorization to accept an item, which during manufacture or after having been submitted for inspection, is found to depart from specified requirements, but nevertheless is considered suitable for use “as is” or after repair by an approved method.