EOSDIS Maintenance and Development Project

Release 7.11
Mission Operation Procedures
for the EMD Project

Revision 02

July 2006

Raytheon Company
Upper Marlboro, Maryland
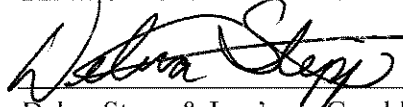
Release 7.11 Mission Operation Procedures
for the EMD Project

Revision 02

July 2006

Prepared Under Contract NAS5-03098
CDRL Item 23

RESPONSIBLE AUTHOR

_____   Date
Debra Stepp & Lay'wan Gamble
EOSDIS Maintenance and Development Project

RESPONSIBLE OFFICE

_____   Date
Mary Armstrong, Program Manager
EOSDIS Maintenance and Development Project

Raytheon Company
Upper Marlboro, Maryland

611-EMD-001, Rev. 02

This page intentionally left blank.

Preface

This document is a formal contract deliverable. It requires Government review and approval within 45 business days. Changes to this document will be made by document change notice (DCN) or by complete revision.

Any questions should be addressed to:

Data Management Office
The EMD Project Office
Raytheon Company
1616 McCormick Drive
Upper Marlboro, Maryland 20774-5301

Revision History

Document Number	Status/Issue	Publication Date	CCR Number
611-EMD-001	Original	April 2004	04-0220
611-EMD-001	Revision 01	July 2005	05-0319
611-EMD-001	Revision 02	July 2006	06-0376

This page intentionally left blank.

Abstract

This document, Mission Operation Procedures for the EMD Project, provides DAAC procedures that assign and describe operators, engineers, operations support, administration and management staff actions required to configure, maintain and operate the ECS applications at maturity. The DAAC portion of this document contains system-level standard procedures that can be modified at the DAACs during subsequent training, operations exercises and procedure review activities to reflect desired uniqueness. The objectives of the current release of the system are to provide capability to support the ingest and archive of raw data obtained from instruments on Earth Observing System (EOS) satellites [e.g., the EOS AM Mission spacecraft 1, morning equator crossing spacecraft series (Terra (AM-1)), EOS PM Mission spacecraft 1 and the afternoon equator crossing spacecraft series (Aqua (PM-1))]. Other capabilities provided by the current release include processing the data obtained, distributing raw or processed data as requested, quality assurance of processed data, supporting communication networks, and systems monitoring via interfaces with the ECS operations staff.

Keywords: operations, DAACs, SMC, EOC, mission support, operation procedures, EOS, Terra (AM-1), Aqua (PM-1), Aura, software, integration, test, SSI&T, Version 0, Release 7.11

This page intentionally left blank.

Contents

1. Introduction

2. Related Documentation

3. System Administration

4. Database Administration

5. Security Services

6. Network Administration

7. System Monitoring

8. Problem Management Procedures

9. Configuration Management Procedures

10. Metadata Administration

11. Production Rules

12. Resource Planning

13. Production Planning

14. Production Processing

15. Quality Assurance

16. Ingest

17. Archive Procedures

18.   Data Distribution

19. User Services

20. Library Administration

21. COTS Hardware Maintenance

22. Software Maintenance

23. Property Management

24. Installation Planning

25. COTS Training

26. Science Software Integration and Test (SSI&T)

27. Inventory Logistical Management (ILM)

28. Maintenance of Configuration Parameters

List of Figures

List of Tables

Appendix A. Additional Material

Abbreviations and Acronyms

1. Introduction

This document, Release 7.11 Mission Operation Procedures for the Earth Observing System Data and Information System (EOSDIS) Maintenance and Development (EMD) Project, provides procedures to configure maintain and operate the EOSDIS Core System (ECS).

1.1 Identification

This document meets the milestone specified as Contract Data Requirements List (CDRL) Item 23, under contract NAS5-03098. It reflects the system as delivered at Release 7.11.

1.2 Scope

The scope of this document is directed to Distributed Active Archive Center (DAAC) operations activities to support the Release 7.11 ECS system. Both procedures and instructions are identified. Operations procedures are defined as the step-by-step commands or on-line procedures needed to perform a function. The Operations Instructions are the off-line procedures or directives for performing administrative, operations, management, or operations support activities (e.g., Configuration Management, Problem Management, or Quality Assurance).

1.2.1 On-Site Procedures Tailoring Guide

Each DAAC may modify these procedures and instructions to accommodate site-specific operations requirements. Such documentation should be versioned and dated in MS Word format with a master copy forwarded to the following address:

EMD Training
The EMD Project Office
Raytheon Company
1616 McCormick Drive
Upper Marlboro, MD 20774-5301

For specifics on authoring, formatting, importing, exporting and maintenance of procedures and instructions see Chapter 20, Library Administration.

1.3 Purpose

The purpose of this document is to identify the procedures and instructions to operate and maintain Release 7.11 systems. In addition, DAAC staff responsibilities are identified. The DAAC operations staff is comprised of operators, engineers, as well as operations support, administration and management staff personnel.

This document will also be used as a training aid for operations staff who are located at the sites. The operations procedures and operations instructions were derived from, and are intended to be

consistent with, the system functions and capabilities specified in the system design specifications.

1.4 Status and Schedule

This document is to be delivered on an annual basis. Updates are made to reflect new system releases. Changes are submitted through established configuration management procedures, such as configuration change requests or published revisions known as interim updates published to the web site at http://edhs1.gsfc.nasa.gov/ at an "Interim Updates" link on the abstract page for this document (611-EMD-001).

1.5 Organization

The contents subsequent to this first section are presented as follows:

- Section 2   **Related Documentation.** Lists documents that drive, support or expand on the material in this manual.

- Section 3   **System Administration.**   Identifies the operations procedures and/or operations instructions for system administration activities, such as backup and restore, log maintenance, user account administration, and workstation installation.

- Section 4   **Database Administration.**   Identifies the operations procedures and/or operations instructions for database administration activities, such as product installation, disk storage management, login and privileges administration, database validation, backup and recovery, database configuration, tuning and performance monitoring.

- Section 5   **Security Services.** Identifies the operations procedures and/or operations instructions for security services activities, such as user authentication and authorization, data access control, network services monitoring, password protection, file modification monitoring.

- Section 6   **Network Administration.**   Identifies the operations procedures and/or operations instructions for network administration activities, such as network and system configuration monitoring, and network services monitoring.

- Section 7   **System Monitoring.** Identifies the operations procedures and/or operations instructions for network system monitoring, such as problem monitoring and resolution.

- Section 8   **Problem Management.**   Identifies the operations procedures and/or operations instructions for submitting trouble tickets and for processing and resolving trouble ticket submissions.

- Section 9   **Configuration Management.** Identifies the operations procedures and/or operations instructions for configuration management activities, such as Configuration Control Board (CCB) support, configuration item identification, submission and processing of configuration change requests (CCRs), configuration status accounting, configuration audits, data management, operational database maintenance, software transfer and installation.

- Section 10   **Metadata Administration.** Identifies the operations procedures and/or operations instructions for metadata administration activities, such as establishing collections, populating the database, and specifying Earth Science Data Type (ESDT) services.

- Section 11   **Production Rules.** This section is intended to explain the production rules governing the use of product generation executives (PGEs) in the system. This section addresses the syntax for specifying production rules.

- Section 12   **Resource Planning.** Identifies the operations procedures and/or operations instructions for resource planning activities for non-production (ground) events.

- Section 13   **Production Planning.** Identifies the operations procedures and/or operations instructions for production planning activities for production jobs, resource prioritization, and scheduling.

- Section 14   **Production Processing.** Identifies the operations procedures and/or operations instructions to support data processing activities.

- Section 15   **Quality Assurance.** Identifies the operations procedures and/or operations instructions to perform DAAC manual non-science quality assurance activities, such as visualization of science data products and updating quality assurance metadata.

- Section 16   **Ingest.** Identifies the operations procedures and/or operations instructions to support data acquisition.

- Section 17   **Archive Procedures.** Identifies the operations procedures and/or operations instructions for archiving activities, such as archive repository maintenance, fault monitoring and notification, and temporary data storage.

- Section 18   **Data Distribution.** Identifies the operations procedures and/or operations instructions to support data distribution activities, such as media operations and product shipment.

- Section 19   **User Services.** Identifies the operations procedures and/or operations instructions to support user services activities to address user requests for data.

- Section 20 **Library Administration.** Identifies the operations procedures and/or operations instructions to support librarian administration activities, such as change package preparation and distribution, master document control and maintenance.

- Section 21 **COTS Hardware Maintenance.** Identifies the operations procedures and/or operations instructions for preventive and corrective maintenance activities of commercial off-the-shelf (COTS) hardware for the EMD project.

- Section 22 **Software Maintenance.** Identifies the operations procedures and/or operations instructions to support maintenance activities for COTS software, custom software, and science software.

- Section 23 **Property Management.** Identifies the operations procedures and/or operations instructions for the receipt, control, and accountability of EMD property at all affected sites.

- Section 24 **Installation Planning.** Identifies the operations procedures and/or operations instructions to support installation planning activities for conducting site surveys, ensuring that site preparations/coordination are completed on schedule, facilitating receipt and installation of the hardware.

- Section 25 **COTS Training.** Identifies the operations procedures and/or operations instructions to support COTS training activities, such as training request processing, training coordination, training scheduling, and training record maintenance.

- Section 26 **Science Software Integration and Test (SSI&T).** Identifies the operations procedures and/or operations instructions to support science software integration and test activities.

- Section 27 **Inventory Logistical Management (ILM).** ILM helps the operations staff at the DAACs, EOC, and SMC to maintain records that describe all inventory components and their assembly structures and interdependencies. The database maintained by this tool keeps chronological histories (a record of the transactions) of receipt, installation, and relocation of inventory items. There is a license management section and general updates to work order processes, forms, and report formats.

- Section 28 **Maintenance of Configuration Parameters.** These procedures describe the overall maintenance of the system configuration parameters baseline for custom software and hardware, including patches, database, operating systems, COTS software, and networks.

- Appendix A **Additional Material.** Contains examples of Object Description Language (ODL) files used by various instrument teams, to illustrate concepts discussed in Section 26.

- **Abbreviations and Acronyms.** Identifies abbreviations and acronyms used throughout this document.

This page intentionally left blank.

2. Related Documentation

2.1 Parent Documents

The parent documents are the documents from which the Mission Operation Procedures' scope and content are derived.

423-41-01	Goddard Space Flight Center, EOSDIS Core System (ECS) Statement of Work
423-46-03	EMD Task 101 Statement of Work For ECS SDPS Maintenance
423-46-02	Contract Data Requirements Document for EMD Task 101 ECS SDPS Maintenance

2.2 Applicable Documents

The following documents are referenced within the Mission Operation Procedures document, or are directly applicable, or contain policies or other directive matters that are binding upon the content of this volume.

420-05-03	Goddard Space Flight Center, Earth Observing System (EOS) Performance Assurance Requirements for the EOSDIS 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 (ECS F&PRS)
423-46-01	Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) Science Data Processing System (EMD F&PRS)

2.3 Information Documents

2.3.1 Information Documents Referenced

The following documents are referenced herein and amplify or clarify the information presented in this document. These documents are not binding on the content of the Mission Operation Procedures for the EMD Project.

105-EMD-001	Property Management Plan for the EMD Project
110-EMD-001	Configuration Management Plan for the EMD Project
170 EMD 003	A Data Formatting Toolkit for Extended Data Providers to NASA's Earth Observing System Data and Information System (V5.0)

313-EMD-001	Release 7.10 Internal Interface Control Document for the EMD Project
333-EMD-001	Release 7 SDP Toolkit Users Guide for the EMD Project
420-EMD-001	Release 7 Implementation Earth Science Data Model for the EMD Project
500-EMD-001	Terra Spacecraft Ephemeris and Attitude Data Preprocessing
500-EMD-002	Aqua Spacecraft Ephemeris and Attitude Data Preprocessing
500-EMD-003	Aura Spacecraft Ephemeris and Attitude Data Preprocessing
609-EMD-001	Release 7.11 Operations Tools Manual for the EMD Project
162-TD-001	Science Software I&T Operational Procedures for the ECS Project (a.k.a., The Green Book)
910-TDA-003	COTS Software Version Baseline Report
910-TDA-005	Site-Host Map Report
910-TDA-021	SYBASE SQLServer 11.0.x ALL DAAC Database Configurations
910-TDA-022	Custom Code Configuration Parameters for ECS
910-TDA-023	Critical COTS Software List
910-TDA-030	COTS [Software] Where Used Report
914-TDA-337	Whats Up Professional 2006 (WhatsUp Gold 8.0) Maintenance Upgrade for the EMD Project: Release Notes
921-TDx-001	DAAC LAN Topology
921-TDx-002	[DAAC] Hardware/Network Diagram
921-TDx-003	IP Address Assignment (DAAC Hosts)
921-TDx-004	IP Address Assignment (DAAC Network Hardware)
921-TDx-005	Dual-Homed Host Static Routes
CM-004	EMD Project Instruction: CCB Change Control Process
CM-1-032-1	EMD Project Instruction: COTS and Custom Software Preparation and Delivery
CM-045	EMD Project Instruction: EMD Software Build Process
DM-002	EMD Project Instruction: Data Identification Numbering
MIL-HDBK-263B	Electrostatic Discharge Control Handbook for Protection of Electrical and Electronic Parts, Assemblies, and Equipment (Excluding Electrically Initiated Explosive Devices) (Metric)

| MIL-STD-1686C | Department of Defense Standard Practice: Electrostatic Discharge Control Program for Protection of Electrical and Electronic Parts, Assemblies and Equipment (Excluding Electrically Initiated Explosive Devices) |

2.3.2 Information Documents Not Referenced

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 Mission Operation Procedures for the EMD Project.

290-004	Goddard Space Flight Center, Internet Protocol Operational Network (IONet) Access Protection Policy and Requirements
423-10-21	Earth Science Data and Information System Project Configuration Management Procedures
423-16-01	Data Production Software and Science Computing Facility (SCF) Standards and Guidelines
305-EMD-001	Release 7.11 Segment/Design Specification for the EMD Project
311-EMD-001	Release 7.11 Data Management Subsystem (DMS) Database Design and Database Schema Specifications for the EMD Project
311-EMD-002	Release 7.11 INGEST (INS) Database Design and Schema Specifications for the EMD Project
311-EMD-003	Release 7.11 Planning and Data Processing Subsystem Database Design and Schema Specifications for the EMD Project
311-EMD-004	Release 7.11 Science Data Server Database Design and Schema Specifications for the EMD Project
311-EMD-005	Release 7.11 Storage Management and Data Distribution Subsystems Database Design and Database Schema Specifications for the EMD Project
311-EMD-006	Release 7.11 Subscription Server Database Design and Schema Specifications for the EMD Project
311-EMD-007	Release 7.11Systems Management Subsystem Database Design and Schema Specifications for the EMD Project
311-EMD-008	Release 7.11 Registry Database Design and Schema Specifications for the EMD Project
311-EMD-010	Release 7.11 NameServer Database Design and Schema Specifications for the EMD Project
311-EMD-011	Release 7.11 Order Manager Database Design and Database Schema Specifications for the EMD Project

311-EMD-012	Release 7.11 Spatial Subscription Server (SSS) Database Design and Schema Specifications for the EMD Project
311-EMD-013	Release 7.11 Data Pool Database Design and Schema Specifications for the EMD Project
508-EMD-001	ACRONYMS for the EOSDIS Maintenance and Development (EMD) Project
905-TDA-001	EMD System Baseline Specification
914-TDA-331	Solaris_8_OS_patches_0905
920-TDx-001	Hardware-Design Diagram
920-TDx-002	Hardware-Software Map
920-TDx-009	DAAC HW Database Mapping
920-TDx-019	Hosts' Custom Code Baseline
152-TP-003	Glossary of Terms for the EOSDIS Core System (ECS) Project
FB9401V2	EOSDIS Core System Science Information Architecture
NPR 1600.1	NASA Procedural Requirements: NASA Security Program Procedural Requirements
NPR 2810.1	NASA Procedural Requirements: Security of Information Technology
OMB Circular A-130	Office of Management and Budget, Management of Federal Information Resources

3. System Administration

This section covers the procedures necessary for the System Administrator (SA) and/or Operator (OPR) to manage and operate the system.

Detailed procedures for tasks performed by the System Administrator and/or Operator are provided in the sections that follow. The procedures assume that the administrator and/or operator is authorized and has proper access privileges to perform the tasks (i.e., root) and that the SA and/or OPR has been properly trained in all aspects of the system.

Each procedure outlined will have an **Activity Checklist** table that will provide an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one   -   **Order** shows the order in which tasks should be accomplished.

Column two   -   **Role** lists the Role/Manager/Operator responsible for performing the task.

Column three - **Task** provides a brief explanation of the task.

Column four   -   **Section** provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five   -   **Complete?** is used as a checklist to keep track of which task steps have been completed.

Table 3-1 is an Activity Checklist that provides an overview of the overall system administration processes and who performs them.

Table 3-1. System Administration - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Using Secure Shell	(I) 3.1	
2	OPR	System Startup/Shutdown	(I) 3.2	
3	OPR	System Backup and Restore	(I) 3.3	
4	SA	User Administration	(I) 3.4	
5	SA	Security	(I) 3.5	

For procedures outlined in this section, there are corresponding **QUICK STEP** procedures immediately following in this chapter. The **QUICK STEP** procedures are designed for persons who have *prior training or are experienced system administrators with prior system administration experience*. The **QUICK STEP** procedures should be used by *experienced persons ONLY*.

3.1 Using Secure Shell

Secure Shell (ssh) is a set of programs that greatly improve network security. The primary need for it on the system is to allow secure, interactive access to the DAACs without needing burdensome procedures and mechanisms and additional hardware.

Secure in this context means not sending passwords "in the clear" so that hackers may intercept them and also provides encryption of the entire session.

Secure Shell is to be used for any inter-access among system platforms and between DAACs.

Table 3.1-1, below, provides an Activity Checklist for secure shell tasks.

Table 3.1-1. Secure Shell - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Initiate sshsetup	(I) 3.1.1	
2	OPR	Set up remote access ssh	(I) 3.1.2	
3	OPR	Change your passphrase	(I) 3.1.3	
4	OPR	Log in to System Hosts	(I) 3.1.4	

3.1.1 Initiate SSHSETUP

Most users will start from the same host whether from an X terminal, a UNIX workstation or a PC. Prior to executing ssh commands, use **setenv DISPLAY <IP address>:0.0** at your local host. To ensure system security, do not use the **setenv DISPLAY** command on subsequent hosts accessed via ssh. The process is started by running the sshsetup script, which will enable ssh to other hosts from which one may use the same home directory. The only thing you need to do before executing the script is to pick a good passphrase of at least 10 characters. You can, and should, use spaces and multiple words with numbers and misspellings and special characters. Note that passwords are NOT echoed back to the screen.

Table 3.1-2 presents (in a condensed format) the steps required to initiate sshsetup. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Login to your normal UNIX workstation where your home directory resides.

2   To initiate Secure Shell setup enter:

      sss

- You will see an information statement:

  Use a passphrase of at least 10 characters which should include numbers or special characters and MAY include spaces

3 At the prompt "**New passphrase:**" enter:

<passphrase>

4 At the prompt "**Retype new passphrase:**" re-enter:

<passphrase>

- You will then see:

  Initializing random number generator...
  Generating p: Please wait while the program completes ...
  %

- This establishes the .ssh sub-directory in your **<username>/home directory**, creates the local ssh key, and creates the necessary files.

Table 3.1-2. Initiate SSHSETUP - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**/usr/local/bin/sss or sss**	**enter text, press Enter**
2	**<passphrase>**	**enter text, press Enter**
3	**<passphrase>**	**enter text, press Enter**

3.1.2 Set Up Remote Access SSH

If you need to access a host with a different home directory, you will need to run the sshremote script. This script sets up the destination host with the new set of keys and transfers the source (local) key to the destination and the destination key to the source. You must have an existing account on the remote host.

Table 3.1-3 presents (in a condensed format) the steps required to set up remote access shell (ssh). If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 Login into your normal UNIX workstation where your home directory resides.

2 To initiate Secure Shell remote setup enter:

ssr

- You will see the following prompt: Remote user name: (**Default: <name>**):
  Note: You must have an account on the remote system first.

  You have a local passphrase. Do you want to setup for:
  1   VATC

2   EDF
                    3   GSFC DAAC
                    4   SMC
                    5   GSFC M and O
                    6   EDC DAAC
                    7   EDC M and O
                    8   LaRC DAAC
                    9   LaRC M and O
                    10   NSIDC DAAC
                    11   NSIDC M and O
                    x   Exit from script
                    Select:

3   At the "**Select**" prompt, enter:

            <host number>

            • You will receive a prompt similar to the following:

               Enter the passphrase for the remote system: <password>

4   At the prompt "**Enter existing passphrase for Remote System:**" enter:

            <passphrase>

            • A prompt similar to the following will be displayed:

               Working…

NOTE: The ssh keys at remote sites can be different from the local-host ssh key.

Table 3.1-3. Set Up Remote Access SSH - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**/usr/local/bin/ssr or ssr**	**enter text, press Enter**
2	**remote user name: <host number>** (at the "**Select**" prompt)	**enter text, press Enter**
3	**<password>**	**enter text, press Enter**
4	**<passphrase>**	**enter text, press Enter**
5	**exit**	**enter text, press Enter**

3.1.3 Change Your Passphrase

Another script has been developed to change your passphrase on the local host and then transfer the key to the other environments. The ssh keys for remote hosts will have to be changed separately. Use the following procedure to change your passphrase:

Table 3.1-4 presents (in a condensed format) the steps required to change your secure shell passphrase. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Login to your normal UNIX workstation where your home directory resides.

2   To initiate passphrase change enter:

ssp

- You will see the following type of information statement:

    Use a passphrase of at least 10 characters which should include numbers or special characters and MAY include spaces

3   At the prompt "**Old passphrase:**" enter:

<old passphrase>

4   At the prompt "**New passphrase:**" enter:

<new passphrase>

5   At the prompt "**Retype new passphrase:**" re-enter:

<new passphrase>

- You will then see an information prompt similar to the following:

    ssh-keygen will now be executed. Please wait for the prompt to Return! /home/jdoe/.ssh/authorized_keys permissions have already been set. %

Table 3.1-4. Change Your Passphrase - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	/usr/local/bin/ssp or ssp	enter text, press Enter
2	<old passphrase>	enter text, press Enter
3	<new passphrase>	enter text, press Enter
4	<new passphrase>	enter text, press Enter

3.1.4 Log in to System Hosts

Logging in to system hosts is accomplished from a UNIX command line prompt. Table 3.1-5 presents (in a condensed format) the steps required to log in to system hosts. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the

system, or have not performed this task recently, you should use the detailed procedures that follow.

NOTE: Commands in Steps 1 and 2 are typed at a UNIX system prompt.

1   In the terminal window (at the command line prompt) start the log-in to the appropriate host by entering:

ssh <host name>

- The **-l** option can be used with the ssh command to allow logging in to the remote host (or the local host for that matter) with a different user ID. For example, to log in to x0ins01 as user cmops enter:

    /ssh -l cmops x0ins01

- Depending on the set-up it may or may not be necessary to include the path (i.e., /usr/local/bin/) with the ssh command. Using ssh alone is often adequate. For example:

    ssh x0ins01

    - or -

    ssh -l cmops x0ins01

- Examples of Sun Consolidation External Server host names include **e0ins01, g0ins01**, **l0ins01**, and **n0ins02**.
- Examples of Sun Consolidation Internal Server host names include **e0acs11**, **g0acs11**, **l0acs03**, and **n0acs04**.
- If you receive the message, "**Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?**" enter **yes** ("**y**" alone will not work).
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Passphrase for key '**user@localhost**'** appears; continue with Step 3.
- If you have not previously set up a secure shell passphrase, go to Step 4.

2   If a prompt to **Passphrase for key '**user@localhost**'** appears, enter:

<passphrase>

- If a command line prompt is displayed, log-in is complete.
- If the passphrase is unknown, press **Return/Enter**, which should cause a **<user@remotehost>'s password:** prompt to appear (after the second or third try if not after the first one), then go to Step 5.
- If the passphrase is entered improperly, a **<user@remotehost>'s password:** prompt should appear (after the second or third try if not after the first one); go to Step 5.

3   If a prompt for *<user@remotehost>***'s password:** appears, enter:

<password>

- A command line prompt is displayed.
- Log-in is complete.

Table 3.1-5. Log in to System Hosts - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**/usr/local/bin/ssh <host name> or ssh <host name>**	**enter text, press Enter**
2	**<passphrase>** (if applicable)	**enter text, press Enter**
3	**<password>** (if applicable)	**enter text, press Enter**

3.2 System Startup and Shutdown

The Startup and Shutdown processes begin when it has been determined by the DAAC Operations Supervisor or his designee that it is necessary to stop or start the system. The least impacting method is determined and users are appropriately notified.

When determining the least impacting way to perform the startup or shutdown, the OPR, along with the Operations Supervisor takes into consideration whether only specific server software packages would need to be started/stopped or an entire system startup/shutdown is required.

Once these steps have been taken, the shutdown or startup is performed.

The Activity Checklist table that follows provides an overview of the startup and shutdown processes.

Table 3.2-1. Startup/Shutdown - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPS Supervisor	Determine that Startup/Shutdown is necessary	(I) 3.2	
2	OPR	Determine the Least Impacting Way to Perform the Startup/Shutdown	(I) 3.2	
3	OPR	Notify Those Affected by the Startup/Shutdown	(I) 3.2	
4	SA	Perform the Startup/Shutdown	(P) 3.2	

3.2.1 Startup

Startup means that power to the system is restored and the system is being taken to a fully useable and operational state.

3.2.1.1 Start Up a Machine

Table 3.2-2 presents (in a condensed format) the steps required to start up a machine. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Power on peripherals before powering on each CPU.

2   Monitor boot-up activity for each system on that system's monitor.

3   Once each system has booted without error, repeat Steps 1 and 2 for each additional machine (if any).

Table 3.2-2. Start Up a Machine - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	(No entry)	power on peripherals before powering on each CPU
2	(No entry)	monitor boot-up activity for each system on that system's monitor
3	(No entry)	repeat Steps 1 and 2 for each additional machine (if any)

3.2.1.2 Perform Cold Startup by Subsystem

A cold startup means that power to the system has been previously powered off and the system(s) is being restarted from this cold state. The System Startup process begins after a previously completed shutdown, either scheduled or emergency. The System Startup is done in sequential order by subsystem. The startup sequence is predetermined by the SA. The sequence may be affected by new software releases.

This procedure assumes that the OPR has been properly trained to start up all aspects of the system and that the system is currently powered off (due to a normal or emergency shutdown).

The procedure assumes that the Startup has been scheduled well in advance, all planning involved has been concluded well in advance and all other Distributed Active Archive Centers (DAACs) have been notified of the system returning to an on-line state.

This section explains how to perform a cold system startup by subsystem. The sequence of the execution of the steps below is VERY IMPORTANT. The following order for host startup is recommended:

- x0css02
- DNS Master Server
- NIS Master

- CSS Server
- x0mss0x
- ClearCase Server
- x0ins01
- Interface Server
- x0mshxx
- x0acs0x
- PDPS
- Other Servers

Table 3.2-3 presents (in a condensed format) the steps required to perform a cold system startup. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

NOTE: The sequence of booting the machines is IMPORTANT:

1 Determine which machines host the following functions (some machines may perform multiple functions).
- Domain Name Server (DNS) Master
- Name Information Server (NIS) Master
- Mail Hub Server(s)
- Automount Servers
- ClearCase Server
- Communication Subsystem (CSS)
- Other License Servers
- Sybase ASE Servers
- Data Server Subsystem (DSS)
- Planning & Data Processing System (PDPS)
- Client Subsystem (CLS)

NOTE: Start up each machine in accordance with the **Start Up a Machine** procedure (Section 3.2.1.1)

2 Start up the DNS Master, the NIS Master, and the CSS server.
- Once that system has booted without error, proceed to Step 3.

3 Power on the ClearCase server(s).
- Once the systems(s) have booted without error, proceed to Step 4.

4 Power on the Interface server(s).
- Once the system(s) have booted without error, proceed to Step 5.

5 Power on the MSS server(s).
- Once the system(s) have booted without error, proceed to Step 6.

6 Power on the DSS server(s).
- Once the system(s) have booted without error, proceed to Step 7.

7 Power on the Ingest server(s).
- Once the system(s) have booted without error, proceed to Step 8.

8 Power on the PDPS server(s).
- Once the system(s) have booted without error, proceed to Step 9.

9 Power on the Client and Data Management server(s).

Table 3.2-3. Cold Startup by Subsystem - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**(No entry)**	**determine which machines host which functions**
2	**(No entry)**	**start up the DNS Master, the NIS Master, and the CSS server** (Use procedure in Section 3.2.1.1)
3	**(No entry)**	**start up the ClearCase server(s)** (Use procedure in Section 3.2.1.1)
4	**(No entry)**	**start up the Interface server(s)** (Use procedure in Section 3.2.1.1)
5	**(No entry)**	**start up the MSS server(s)** (Use procedure in Section 3.2.1.1)
6	**(No entry)**	**start up the DSS server(s)** (Use procedure in Section 3.2.1.1)
7	**(No entry)**	**start up the Ingest server(s)** (Use procedure in Section 3.2.1.1)
8	**(No entry)**	**start up the PDPS server(s)** (Use procedure in Section 3.2.1.1)
9	**(No entry)**	**start up the Client and Data Management server(s)** (Use procedure in Section 3.2.1.1)

3.2.1.3 Perform Warm Startup by Subsystem

A warm startup means the system has been previously powered on, but the system(s) is not fully operational, either the system has had some service performed (i.e. single-user mode) or is being rebooted to correct some minor malfunction. The System Startup is done in sequential order by subsystem. This startup sequence is predetermined by the software dependencies.

The order of the re-boot is contingent on software dependencies per site.

If the NIS server service has been interrupted, the users will automatically be transferred to a backup server. Once the faulty server(s) has been repaired, re-establish connection with the primary NIS server by rebooting the Backup Server; the users would then be transferred back to the primary NIS server.

Table 3.2-4 presents (in a condensed format) the steps required to perform a warm system startup. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 Determine which machines perform the following functions:
- Domain Name Server (DNS) Master
- Name Information Server (NIS) Master
- Mail Hub Server(s)
- Automount Servers
- ClearCase Server
- Communication Subsystem (CSS)
- Other License Servers
- System Management Subsystem
- Sybase ASE Servers
- Data Server Subsystem (DSS)
- Planning & Data Processing System (PDPS)
- Client Subsystem (CLS)

2 Determine which machine is currently down.

3 Determine the interoperability dependencies among the machines.

NOTE: Start up each machine in accordance with the **Start Up a Machine** procedure (Section 3.2.1.1)

4 Turn on machines in an order consistent with the dependencies.

NOTE: In addition to warm system startup/reboot sequences, system servers that use the Sybase ASE server may need to be bounced whenever the ASE server is bounced. At present, this is certainly the case for all STMGT servers. That is, if the Sybase ASE server is stopped and restarted for any reason, all STMGT servers need to be stopped and restarted once the Sybase ASE server has come back on-line.

Table 3.2-4. Perform Warm Startup by Subsystem - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	(No entry)	determine software dependencies
2	(No entry)	reboot independent server(s)
3	(No entry)	reboot dependent server(s)

3.2.1.4 Update leapsec.dat and utcpole.dat Files

In order to ensure proper operation of Product Generation Executives (PGEs), two files must be updated weekly with data transferred from the U.S. Naval Observatory. These files are **${PGSHOME}/database/common/TD/leapsec.dat** and **${PGSHOME}/database/common/CSC/utcpole.dat**. The updating of these files is accomplished by executing **leapsec_update.sh** and **utcpole_update.sh** in the **/usr/ecs/OPS/CUSTOM/TOOLKIT/toolkit/bin/sgi64_daac_f77** directory on each Science Processor with root privileges. If the tasks have not been set up to be accomplished via cron job, they must be done manually.

To manually update leapsec.dat and utcpole.dat files, execute the procedure steps that follow:

Table 3.2-5 presents (in a condensed format) the steps required to perform a normal system shutdown. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

NOTE: The user account performing the update must have write permission in the Toolkit directories.

1 Log in to one of the Science Processor machines (as a user who has write permission in the Toolkit directories).
- Examples of Science Processor host names include **e0spg11**, **g0spg01**, and **l0spg11**.

2 At the UNIX command line prompt enter:

cd /usr/ecs/*<mode>*/CUSTOM/TOOLKIT/toolkit/bin/sig64_daac_f77

3 At the UNIX command line prompt enter:

source pgs-env.csh

4   At the UNIX command line prompt enter either of the following commands:

update_leapsec.sh

- or -

update_utcpole.sh

- **update_leapsec.sh** is used for updating the leap seconds file.
- **update_utcpole.sh** is used for updating the leap seconds file.

5   Enter responses to program prompts as appropriate.

6   Repeat Steps 4 and 5 to update the other file.

7   Repeat Steps 1 through 6 for each additional Science Processor host (if any).

Table 3.2-5. Update leapsec.dat and utcpole.dat Files - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**(No entry)**	**log in to one of the Science Processor machines as a user who has write permission in the Toolkit directories**
2	**cd /usr/ecs/<mode>/CUSTOM/TOOLKIT/toolkit/bin/sig64_daac_f77**	**enter text, press Enter**
3	**source pgs-env.csh**	**enter text, press Enter**
4	**update_leapsec.sh** - or - **update_utcpole.sh**	**enter text, press Enter**
5	responses to program prompts (as appropriate)	**enter text, press Enter**
6	**(No entry)**	**repeat Steps 4 and 5 to update the other file**
7	**(No entry)**	**repeat Steps 1 through 6 for each additional Science Processor host (if any)**

3.2.2 Shutdown

Shutdown means that the system is being removed from a fully useable and operational state and possibly, power to the system will be terminated. The types of shutdown would vary depending upon circumstances (i.e., shut down to single-user mode; shut down to power off; etc.)

3.2.2.2 Perform Normal Shutdown by Subsystem

The Normal System Shutdown process is performed at the discretion of the SA usually for a scheduled repair. The system shutdown is normally performed in reverse order of the system startup.

The procedures assume that the Shutdown has been scheduled well in advance, all planning involved has been concluded well in advance and the OPR/SA has been properly trained to shut down all aspects of the system.

This section explains how to perform a normal system shutdown by subsystem.

Table 3.2-6 presents (in a condensed format) the steps required to perform a normal system shutdown. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

NOTE: The servers should be shut down in the reverse order of the startup.

1 Determine subsystem and server dependencies.
  - The NIS server must be the last system to be shut down.

2 Shut down each machine in order as described in Steps 3 through 10.

3 Log in to the server as root.

4 Enter root password.

5 At the UNIX prompt enter:

  wall <message>

  - To cross NFS mounts on Sun machines enter:

    wall -a <message>

  - Use the following type of message:

    This machine is being shut down for <reason>. The anticipated length of down time is <number of minutes>. Please save your work and log off now. The machine will be coming down in <number> minutes. We are sorry for the inconvenience.

  - Include your name and appropriate telephone number.

6 At the UNIX prompt enter:

  Ctrl-d

7 Wait at least five minutes.

8   At the UNIX prompt enter:

      shutdown -g\<number of seconds\>   -i0 -y

9   When the system is at the prompt, power off all system peripherals.

10   When all system peripherals have been powered off, power off the CPU.

11   Repeat Steps 3 through 10 on the next machine (as determined in Step 1) to be shut down until all machines have been shut down.

Table 3.2-6. Perform Normal Shutdown by Subsystem - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**(No entry)**	**determine subsystem and server dependencies**
2	**(No entry)**	**log in to the server as root**
3	**wall**	**press Enter**
4	**This machine is being shut down for \<reason\>. The anticipated length of down time is \<number of minutes\>. Please save your work and log off now. The machine will be coming down in \<number\> minutes. We are sorry for the inconvenience.**	**enter text, press Ctrl-d**
5	**(No entry)**	**wait at least five minutes**
6	**shutdown -g\<number of seconds\> -i0 -y** - or - **shutdown now -i0 -y**	**press Enter**
7	**(No entry)**	**power off all peripherals**
8	**(No entry)**	**power off the CPU**
9	**(No entry)**	**repeat Steps 2 through 8 for each machine to be shut down (as determined in Step 1)**

3.2.2.3 Perform an Emergency Shutdown by Subsystem

The Emergency System Shutdown process begins after it is determined that the system may fail during emergency situations (i.e., storms, power outages) by the System Administrator (SA). The Emergency System Shutdown is done in sequential order by subsystem. This shutdown sequence is predetermined by the SA.

The NIS server must be the last system to shut down.

Detailed procedures for tasks performed by the OPR/SA are provided in the sections that follow.

This section explains how to perform an emergency system shutdown by subsystem.

Table 3.2-7 presents (in a condensed format) the steps required to perform an emergency system shutdown. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Determine subsystem and server dependencies.
- The NIS server must be the last system to be shut down.

2   Log in to the server as root.
- The OPR must be logged in as root to perform a shutdown.

3   At the UNIX prompt enter:

sync

- **"sync"** executes the sync system primitive. If the system is to be stopped, sync must be called to insure file system integrity. It will flush all previously unwritten system buffers out to disk, thus assuring that all file modifications up to that point will be saved.

4   At the UNIX prompt enter (again):

sync

5   At the UNIX prompt enter:

halt

6   Once the halt has completed, set the power switches on all associated peripherals at the **off** position.

7   Set the power switch for the CPU at the **off** position.

Table 3.2-7. Perform an Emergency Shutdown by Subsystem - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	**(No entry)**	**determine subsystem and server dependencies**
2	**(No entry)**	**log in to server as root**
3	**sync**	**enter text, press Enter**
4	**sync** [again]	**enter text, press Enter**
5	**halt**	**enter text, press Enter**
6	**off**	**set power switches on all associated peripherals**

Step	What to Enter or Select	Action to Take
7	off	set power switch for CPU
8	(No entry)	repeat Steps 2 through 7 for each machine to be shut down (as determined in Step 1)

3.2.2.4 Perform an Extreme Emergency Shutdown (Sun Machines Only)

In case of EXTREME emergency where time does not allow you to execute the **Perform an Emergency Shutdown by Subsystem** procedure (Section 3.2.2.3), execute the following procedure steps for Sun machines ONLY.

Table 3.2-8 presents (in a condensed format) the steps required to perform an extreme emergency shutdown (Sun machines only). If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Log in to the server as root.

2   Simultaneously press the **L1** or **Stop** key and the **a** key simultaneously.
- The use of **L1-a** does not ensure file system integrity. There is a very high risk of losing data when using this process.

3   Once returned to an **ok** or **>** prompt, set the power switches on all associated peripherals at the **off** position.

4   Set the power switch for the CPU at the **off** position.

Table 3.2-8. Perform an Extreme Emergency Shutdown (Sun Machines Only) -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	(No entry)	log in to server as root
2	L1-a	enter text, press Enter
3	off	set power switches on all associated peripherals
4	off	set power switch for CPU

3.3 System Backup and Restore

Performing regular and comprehensive system backups is one of the most important requirements of system operations. Backups are the insurance that essentially all of the system data is always available. If the system crashes and all disks are damaged, the System Administrator should be able to restore all of the data from the backup tapes. Accordingly, the system includes a software product, Legato Networker, which is set up to perform backups automatically on a regular periodic basis. The backups copy critical data to digital linear tape, and Networker can be configured to make clones or copies of some or all of these data, so that it is possible to store the data (e.g., full system backups) in offsite secure storage.

System Backup is the process of copying the information from the machine, either the entire or partial system, for safe keeping for a specific time period. Restore is the process of returning the data to the machine to allow operation to continue from a specific point in time. The operator must be in the admin list to configure and use Networker. This is not root privilege, although root on the Networker server has admin privileges.

An incremental backup copies to tape all files on a system or subsystem that were created or modified since the previous incremental backup regardless of the backup level. The purpose of an incremental backup is to insure that the most recent edition of a file is readily available in case user error or disastrous system failure causes the file to become corrupt. Incremental backups are scheduled at a time that causes minimal disruption to the users.

Incremental backups are performed automatically according to the schedule set up in the Networker Schedules windows. Incremental backups can also be requested at unscheduled times by completing the Incremental Backup Request Form and submitting it to the OPS Supervisor.

A full system backup is a snapshot of the data on the entire system as of a particular date. The data are stored on tapes that are used to recreate the system in the event of a total system failure. Networker runs the full system backup on a regular schedule, usually weekly.

Refer to local procedures for software backup requirements including offsite backups. The procedural information in this document includes information concerning reconfiguration of the Networker setup for backups, with specific provisions for offsite storage of full backup data. Section 3.3.1 concerns incremental backups. Section 3.3.2 includes specific reference to offsite storage of full backup data. Section 3.3.3 addresses reconfiguration of Networker setup to enable the removal of backup tapes for offsite storage.

3.3.1 Incremental Backup

Non-scheduled incremental backups can be requested at any time by submitting a request through REMEDY for Incremental Backup to the OPS supervisor. The supervisor schedules the request with the operator who performs the incremental backup. Afterwards, the operator notifies the requester and supervisor that the incremental backup is complete.

The Activity Checklist in Table 3.3-1 provides an overview of the incremental backup processes. Note: This is for manual backup outside Networker's automatic backup schedule.

Table 3.3-1. Incremental Backup - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request for Incremental Backup to OPS Supervisor	(I) 3.3.1	
2	OPS Supervisor	Schedule Incremental Backup with operator	(I) 3.3.1	
3	OPR	Perform Incremental Backup	(P) 3.3.1	
4	OPR	Notify Requester and OPS Supervisor when Incremental Backup is Complete	(I) 3.3.1	

Detailed procedures for tasks performed by the operator are provided in the sections that follow.

The procedures assume that the requester's request for an incremental backup has already been approved by DAAC Management. Incremental backups can be requested at any time by submitting a request for Incremental Backup to the OPS supervisor. In order to perform the procedure, the operator must have obtained the following information from the requester:

- Name of machine(s) to be backed up
- Files/directories to be backed up (optional)

Table 3.3-2 presents the steps required to perform a non-scheduled incremental backup. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

NOTE: If you run out of tapes at any time during this procedure, execute the **Labeling Tapes** procedure (Section 3.3.6.1) and then return to this procedure.

1   Access a terminal window logged in to the machine to be backed up.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).

2   To start logging in as root, at the command line prompt enter:

su

- A password prompt is displayed.

3   At the command line prompt enter:

<root password>

- Remember that the root password is case-sensitive.
- You are authenticated as root and returned to the UNIX prompt.

4   Execute the Networker Administrative program by entering:

nwadmin &

- A window opens for the Networker Administrative program.
- You are now able to perform an incremental backup.

5   **Single-click** on **Clients**.

6   **Single-click** on **Client Setup**.

7   **Single-click** on the host being backed up.

8   **Single-click** on (highlight) the group to be backed up.

9   Execute the following menu path from the pull-down menu:

Customize → Schedules

- The **Schedules** window opens.

10   Observe the button corresponding to the current date.
- The letter on the button for a particular date specifies the kind of backup that will be done, unless it is overridden.
    - Letter **i** stands for "incremental" backup.
    - Letter **f** stands for "full" backup.
- If there is an **i** next to the date on the button, go to Step 13.
- If there is an **f** next to the date on the button, go to Step 11.

11   Execute the following menu path from the button corresponding to the current date:

<current date> → Overrides → Incremental

12   **Single-click** on the **Apply** button.

13   To close the **Schedules** window **single-click** and **hold** the option button in the upper left corner of the window to display a menu of window options, **move** the cursor to the **Close** option, then **release** the mouse button.
- The **Schedules** window closes.

14   **Single-click** on the **Group Control** button.
- The **Group Control** window opens.

15   **Single-click** on the **Start** button.
- A **Notice** window opens.

16   **Single-click** on the **OK** button.
- The **Notice** window closes.

- The regularly scheduled backup will still run (even though we are now doing a backup).

17 To close the **Group Control** window **single-click** and **hold** the option button in the upper left corner of the window to display a menu of window options, **move** the cursor to the **Close** option, then **release** the mouse button.
- Status updates appear in the **nwadmin** window.
- When the backup is complete, a **Finished** message is displayed.

18 To quit the Networker Administrative program when the backup is complete execute the following menu path:

File → Exit

- The **nwadmin** window closes.

19 At the UNIX prompt for the machine to be backed up enter:

exit

- Root is logged out.

20 At the UNIX prompt enter:

exit

- You are logged out and disconnected from the machine to be backed up.

Table 3.3-2. Perform Incremental Backup - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the machine to be backed up** (Use procedure in Section 3.2.1)
2	**su**	**enter text, press Enter**
3	**<root password>**	**enter text, press Enter**
4	**nwadmin &**	**enter text, press Enter**
5	**Clients**	**single-click**
6	**Client Setup**	**single-click**
7	**<host being backed up>**	**single-click**
8	**<group to be backed up>**	**single-click**
9	**Customize → Schedules**	**single-click**
10	(No entry)	**observe the button corresponding to the current date; if there is an i next to the date on the button, go to Step 13**
11	**<current date> → Overrides → Incremental**	**single-click**
12	**Apply** button	**single-click**

Table 3.3-2. Perform Incremental Backup - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
13	**Close** option [option button in the upper left corner of the **Schedules** window]	**single-click**
14	**Group Control** button	**single-click**
15	**Start** button	**single-click**
16	**OK** button	**single-click**
17	**Close** option [option button in the upper left corner of the **Group Control** window]	**single-click**
18	**File → Exit** [when the backup is complete]	**single-click**
19	exit	**enter text, press Enter**
20	exit	**enter text, press Enter**

3.3.2 Full Backup

Non-scheduled full backups can be requested at any time by submitting a request for Full Backup to the OPS supervisor. The supervisor schedules the request with the operator who performs the full backup. Afterwards, the operator notifies the requester and supervisor that the full backup is complete. In preparation for offsite storage, it is also necessary to copy the file index to a tape for storage offsite with the system backups.

The Activity Checklist in Table 3.3-3 provides an overview of the full backup processes.

Table 3.3-3. Full Backup - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request for Full Backup to OPS Supervisor	(I) 3.3.2	
2	OPS Supervisor	Schedule Full Backup with Operator	(I) 3.3.2	
3	OPR	Perform Full Backup	(P) 3.3.2	
4	OPR	Notify Requester and OPS Supervisor when Full Backup is Complete	(I) 3.3.2	

Detailed procedures for tasks performed by the operator are provided in the sections that follow.

The procedures assume that the requester's application for a full backup has already been approved by DAAC Management. In order to perform the procedure, the operator must have obtained the following information from the requester:

- Name of machine to be backed up
- Files/directories to be backed up (optional)

Table 3.3-4 presents the steps required to perform a full backup for the requester. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new

to the system, or have not performed this task recently, you should use the following detailed procedure:

NOTE: If you run out of tapes at any time during this procedure, execute the **Labeling Tapes** procedure (Section 3.3.6.1) and then return to this procedure.

1   Access a terminal window logged in to the machine to be backed up.
   - For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).

2   To start logging in as root, at the command line prompt enter:

   su

   - A password prompt is displayed.

3   At the command line prompt enter:

   \<root password>

   - Remember that the root password is case-sensitive.
   - You are authenticated as root and returned to the UNIX prompt.

4   Execute the Networker Backup program by entering:

   nwbackup &

   - A Networker Backup window is displayed.
   - You are now able to perform a full backup.

5   **Single-click** on **Clients**.

6   **Single-click** on **Client Setup**.

7   **Single-click** on the host being backed up.

8   **Single-click** on (highlight) the group to be backed up.

9   If no list of files/directories to be backed up was provided (i.e., the whole machine is to be backed up), in the **Selection** field enter:

   /

   - The character **/** is displayed in the **Selection** field.

10   If no list of files/directories to be backed up was provided, **single-click** on the **Mark** button.
   - A check mark next to **/** indicates that it is designated for backup.
   - Go to Step 14.

11   If names of file(s)/directory(ies) to be backed up were provided, in the directory display **single-click** on a file/directory to be backed up.
- **Drag** the scroll bar with the mouse to scroll the list up and down.
- **Double-click** on a directory name to list its contents.
- To move up a directory level, enter the path in the **Selection** field.

12   **Single-click** on the **Mark** button.
- A check mark next to each selected file indicates that it is designated for backup.

13   Repeat Steps 11 and 12 as necessary to mark all file(s)/directory(ies) to be backed up.

14   **Single-click** on the **Start** button.
- A **Backup Options** window opens.

15   **Single-click** on the **OK** button.
- The **Backup Options** window closes.
- The **Backup Status** window opens providing updates on the backup's progress.

16   After the **Backup Completion Time** message appears in the **Backup Status** window, **single-click** on the **Cancel** button.
- The **Backup Status** window closes.
- The backup is complete.

17   To quit the Networker Backup program when the backup is complete execute the following menu path:

File → Exit
- The **Networker Backup** window closes.

18   To copy the file index to tape for offsite storage, at the UNIX prompt enter:

cp <index filename> <tape destination>

19   Remove the cloned system full backup tapes from the STK jukebox for transport to secure offsite storage.

20   Replace the removed system backup tapes with a set of backup tapes rotated in from secure offsite storage.

21   At the UNIX prompt enter:

exit
- Root is logged out.

22 At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the machine to be backed up.

Table 3.3-4. Perform Full Backup - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the machine to be backed up** (Use procedure in Section 3.2.1)
2	**su**	**enter text, press Enter**
3	**<root password>**	**enter text, press Enter**
4	**nwbackup &**	**enter text, press Enter**
5	**Clients**	**single-click**
6	**Client Setup**	**single-click**
7	**<host being backed up>**	**single-click**
8	**<group to be backed up>**	**single-click**
9	**/** (in the **Selection** field) (to back up the whole machine)	**enter text**
10	**<file(s)/directory(ies)>** (to back up specific file(s)/directory(ies)) (in the directory display)	**single-click**
11	**Mark** button	**single-click**
12	(No entry)	**Repeat Steps 10 and 11 as necessary to mark all file(s)/directory(ies) to be backed up**
13	**Start** button	**single-click**
14	**Cancel** button (after the **Backup Completion Time** message appears in the **Backup Status** window)	**single-click**
15	**File → Exit** [when the backup is complete]	**single-click**
16	**cp <index filename> <tape destination>** (to copy the file index to tape for offsite storage)	**enter text, press Enter**
17	(No entry)	**Remove the cloned system full backup tapes from the STK jukebox**
18	(No entry)	**Replace the removed system backup tapes with a set of backup tapes rotated in from secure offsite storage**
19	**exit**	**enter text, press Enter**
20	**exit**	**enter text, press Enter**

3.3.3 Configuring Networker Setup for Backup Clones for Offsite Storage

Detailed information on configuring Networker may be found in the Legato Networker Administrator's Guide, UNIX Version. During installation at the sites, Networker is configured to schedule and perform automatic incremental and full backups. Therefore, much of the initial setup is complete, including licensing, designation of users who have administrative privileges, specification of clients (computers that contain data to be backed up) and their save sets (increments of data to be backed up), and identification of storage devices. If Networker is not configured to create any clones (duplicate copies of save sets) that can be removed from the storage device and stored at an offsite location, it will be necessary to change the pools (collections of backup tape volumes in the storage device) to implement offsite storage. Specifically, it will be appropriate to specify separate pools for full backups and incremental backups, and for clones of the associated full backup save sets that can be stored off site.

The purpose of off-site storage of backup data is to enable restoration of the system in the unlikely circumstance of a catastrophic event that causes loss of system software. In the face of such an event, in order to return to operational status, once the system hardware and infrastructure have been determined sound or returned to sound condition, it will be necessary to reload the operating system and restore the system from the backup data. This will require not only the offsite backup data, but also access to media containing the operating system software, Legato Networker software, and indexes to the backup data. Accordingly, it is necessary to secure in offsite storage the operating system software media, Networker software media, and media containing the indexes for backup data, as well as the backup data themselves.

To set up the system for off-site backups, it is necessary to create a label template (used by Networker to create internal labels for tapes) and create a clone pool for the tapes to be used in cloning the full backup for offsite storage. When creating any volume pool in Networker, it is necessary to specify the type of data to include on the volumes in the pool; one of the preconfigured selections provided by Networker is Backup Clone, and this may be used for the type of data to include on volumes in the clone pool.

The Activity Checklist in Table 3.3-5 provides an overview of the process to configure Networker to clone full backups, copy the necessary indexes, and ensure offsite storage of the necessary data.

Table 3.3-5. Configure Networker to Enable Offsite Storage - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Set up Label Template	(P) 3.3.3	
2	OPR	Set up Clone Pool	(P) 3.3.3	

Table 3.3-6 presents the steps required to configure Networker for offsite storage (i.e., to set up for cloning full backups). If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the Tape Backup Server.

- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).
- Examples of Tape Backup Server host names include **e0mss04, g0mss07**, **l0mss05**, and **n0mss05**.

2   To execute the Networker Admin program at the command line prompt enter:

nwadmin &

- The **nwadmin** window is displayed.

3   To quit the Networker Backup program when the backup is complete execute the following menu path:

Customize → Label Templates . . . .

- The **Label Templates** window is displayed.

4   **Single-click** on the **Create** button.

- The data fields are cleared and the cursor is displayed in the **Name:** field.

5   In the **Name:** field enter:

Full Clone

- After typing the text, press the **Tab** key to advance to the next field.
- The typed entry is displayed in the **Name:** field and the cursor is displayed in the **Fields:** field.

6   In the **Fields:** field enter:

<Networker host name>.ecs.nasa.gov

- For **<Networker host name>**, use the Tape Backup Server for Networker at the local site (e.g., **e0mss04**, **g0mss07**, **l0mss05**, or **n0mss05**).
- After typing the text, press the **Tab** key to advance to the next field.
- The typed entry is displayed in the **Fields:** field and in the list window immediately below the field.

7   In the **Fields:** field enter:

Full

- Replace the previous entry in the **Fields:** field either by using the mouse to highlight the entry in the field and then typing the relevant text over it, or by clicking at the end of the field and using the backspace key to delete the entry before typing the relevant text.
- After typing the text, press the **Tab** key to advance to the next field.

- The typed entry is displayed in the **Fields:** field and in the list window immediately below the field.

8   In the **Fields:** field enter:

001-999

- Replace the previous entry in the **Fields:** field either by using the mouse to highlight the entry in the field and then typing the relevant text over it, or by clicking at the end of the field and using the backspace key to delete the entry before typing the relevant text.
- After typing the text, press the **Tab** key to advance to the next field.
- The typed entry is displayed in the **Fields:** field and in the list window immediately below the field.

9   In the **Separator:** field **single-click** on the selection button next to the symbol (**.**) to be used as a separator between components of the labels

- Choices are "**.**", "**_**", "**:**", and "**-**"; use the "**.**").
- The button changes color to indicate that it has been selected.

10   **Single-click** on the **Apply** button.

- The **Next:** field displays **<Networker host name>.ecs.nasa.gov.Full.001** to indicate the next label that will be applied.

11   Execute the following menu path:

File → Exit

- The **Label Templates** window is closed.

12   To begin creation of the Full Backup clone pool, in the Networker main window execute the following menu path:

Media → Pools . . . .

- The **Pools** window is displayed.

13   **Single-click** on the **Create** button.

- The cursor is displayed in the **Name:** field.

14   In the **Name:** field enter:

Full Clone

- After typing the text, press the **Tab** key to advance to the next field.
- The typed entry is displayed in the **Name:** field.

15   If necessary, in the **Enabled:** field **single-click** on the **Yes** button.

- The color of the button indicates that it has been selected.

16   **Single-click** on the pull-down arrow to the right of the **Pool Type:** field, **hold** down the **right** mouse button, and **drag** to select **Backup Clone**.

- The selection is displayed in the **Pool Type:** field.

17   **Single-click** on the pull-down arrow to the right of the **Label Template:** field, **hold** down the **right** mouse button, and **drag** to select **Full Clone**.

- The selection is displayed in the **Label Template:** field.

18   In the **Store Index Entries:** field **single-click** on the selection button for **Yes**.

- The selection is displayed in the **Label Template:** field.
- Use the scroll bar on the right side of the window to scroll down if necessary.
- The color of the button changes to indicate that it has been selected.

19   **Single-click** on the **Apply** button.

- **Full Clone** is displayed in the **Pools:** list.

20   In the **Pools:** list, **single-click** on **Full**.

- **Full** is highlighted and data for the Full pool are displayed in the appropriate data fields.

21   In the **Enabled:** field ensure that the **Yes** button has been selected (**single-click** on the button if necessary) so the Full pool can be used for full backups.

- The color of the button indicates whether or not it has been selected.

22   Execute the following menu path in the **Pools:** window:

File → Exit

- The **Pools:** window is closed.

23   Execute the following menu path in the **nwadmin** window:

File → Exit

- The **nwadmin** window is closed.

Table 3.3-6. Configure Networker to Enable Offsite Storage (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the machine to be backed up** (Use procedure in Section 3.2.1)
2	**nwadmin &**	**enter text, press Enter**
3	**Customize → Label Templates . . . .**	**single-click**
4	**Create** button	**single-click**
5	**Full Clone** (in the **Name:** field)	**enter text, press Tab**

Table 3.3-6. Configure Networker to Enable Offsite Storage (2 of 2)

Step	What to Enter or Select	Action to Take
6	**<Networker host name>.ecs.nasa.gov** (in the **Fields:** field)	**enter text, press Tab**
7	**Full** (in the **Fields:** field)	**enter text, press Tab**
8	**001-999** (in the **Fields:** field)	**enter text, press Tab**
9	**.** [dot] (in the **Separator:** field)	**single-click**
10	**Apply** button	**single-click**
11	**File → Exit**	**single-click**
12	**Media → Pools . . . .** (in the Networker main window)	**single-click**
13	**Create** button	**single-click**
14	**Full Clone** (in the **Name:** field)	**single-click**
15	**Yes** button (in the **Enabled:** field) (if necessary)	**single-click** (if necessary)
16	**Backup Clone** (in the **Pool Type:** field)	**right-click, hold**, and **drag**
17	**Full Clone** (in the **Label Template:** field)	**right-click, hold**, and **drag**
18	**Yes** button (in the **Store Index Entries:** field)	**single-click**
19	**Apply** button	**single-click**
20	**Full** (in the **Pools:** list)	**single-click**
21	**Yes** button (in the **Enabled:** field) (if necessary)	**single-click** (if necessary)
22	**File → Exit** (in the **Pools:** window)	**single-click**
23	**File → Exit** (in the **nwadmin** window)	**single-click**

3.3.4 File Restore

From time to time, individual files or groups of files (but not all files) will have to be restored from an Incremental or Full backup tape(s) due to Operator error or system failure. This can be accomplished using the following file restoration procedure.

The File Restore process begins when the requester submits a request to the Operator. The Operator restores the file(s) and notifies the requester when complete.

The Activity Checklist in Table 3.3-7 provides an overview of the file restore process.

Table 3.3-7. File Restore - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request for File Restore to Operator	(I) 3.3.4	
2	OPR	Perform Single- or Multiple-File Restore	(P) 3.3.4.1	
3	OPR	Inform Requester of completion	(I) 3.3.4	

Detailed procedures for tasks performed by the Operator are provided in the sections that follow.

The procedures assume that the requester's application for a file restore has already been approved by the Ops Supervisor. In order to perform the procedure, the Operator must have obtained the following information from the requester:

- Name of machine to be restored
- Name of file(s) to be restored
- Date from which to restore
- User ID of the owner of the file(s) to be restored
- Choice of action to take when conflicts occur. Choices are:
  - Rename current file
  - Keep current file
  - Write over current file with recovered file

3.3.4.1 Perform Single- or Multiple-File Restore

Table 3.3-8 presents the steps required to perform single- or multiple-file restore. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed these tasks recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the machine to be restored.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).

2   At the command line prompt enter:

       cd <path>

- Change directory to the directory containing the Networker Recovery executable (e.g., /data1/COTS/networker).

3   To execute the Networker Recovery program at the command line prompt enter:

       nwrecover &

- A window opens for the Networker Recovery program.
- You are now able to perform the file restoration.

4   In the directory display **single-click** on a file/directory to be restored.
- **Drag** the scroll bar with the mouse to scroll the list up and down.
- **Double-click** on a directory name to list its contents.

5   **Single-click** on the **Mark** button.
- A check mark next to each selected file indicates that it has been designated for restoration.

6   Repeat Steps 4 and 5 as necessary to mark all file(s)/directory(ies) to be restored.

7   Execute the following menu path:

Change → Browse Time

- The **Change Browse Time** window opens.

8   **Single-click** on the date from which to restore.
- Networker automatically goes to that day's or a previous day's backup which contains the file.

9   **Single-click** on the **Start** button.
- The **Conflict Resolution** window opens.

10   In response to the question **Do you want to be consulted for conflicts? single-click** on the **yes** button.

11   **Single-click** on the **OK** button.
- If prompted with a conflict, the following choices of action will be available: **rename current file**, **keep current file**, or **write over current file with recovered file**. Select the requester's choice of action to take when conflicts occur.
- The **Recover Status** window opens providing information about the file restore.
- If all the required tapes are not in the drive, a notice will appear. **Single-click** on the **OK** button in the notice window.
- If prompted for tapes, **single-click** on **Cancel** in the **Recover Status** window and execute the **Indexing Tapes** procedure (Section 3.3.6.2).

12   When a recovery complete message appears, **single-click** on the **Cancel** button.

13   Execute the following menu path:

File → Exit

- The Networker Recovery program quits.

14   At the UNIX prompt enter:

exit

- The owner of the file(s) to be restored is logged out.

15   At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the machine to be backed up.

Table 3.3-8. Perform Single- or Multiple-File Restore - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the machine to be backed up** (Use procedure in Section 3.2.1)
2	**cd <path>**	**enter text, press Enter**
3	**nwrecover &**	**enter text, press Enter**
4	**<file/directory>** (to be restored) (in the directory display)	**single-click**
5	**Mark** button	**single-click**
6	(No entry)	**Repeat Steps 4 and 5 as necessary to mark all file(s)/directory(ies) to be restored**
7	**Change → Browse Time**	**single-click**
8	**<date>** (from which to restore)	**single-click**
9	**Start** button	**single-click**
10	**yes** button (in response to **Do you want to be consulted for conflicts?**)	**single-click**
11	**OK** button	**single-click**
12	**Cancel** button (when recovery is complete)	**single-click**
13	**File → Exit**	**single-click**
14	**exit**	**enter text, press Enter**
15	**exit**	**enter text, press Enter**

3.3.5 Complete System Restore

The Complete System Restore process begins when the requester has determined that a complete system restore is the only way to resolve the problem and has approval from the Operations Supervisor. Once notified of the request, the Operator performs restores of all partitions on the system. Afterwards, the Operator documents and logs all actions in the operator's logbook and notifies the requester and Ops Supervisor that the system restore is complete.

The Activity Checklist in Table 3.3-9 provides an overview of the complete system restore process.

Table 3.3-9. Complete System Restore - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Troubleshoot and Determine that a Complete System Restore is necessary	(I) 3.3.5	
2	OPR	Perform Full System Restore	(P) 3.3.5.1	
3	OPR	Document and Log in operator's log book, and Inform Requester and Ops Supervisor of completion	(I) 3.3.5	

Detailed procedures for tasks performed by the Operator are provided in the sections that follow. The procedures assume that the requester's application for a complete system restore has already been approved by the Ops Supervisor. In order to perform the procedures, the Operator must have obtained the following information about the requester:

- Name of system to be restored
- Date from which to restore

3.3.5.1 Perform Full System Restore

A complete system restore involves restoring all partitions on that system.

Table 3.3-10 presents the steps required to perform full system restore. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed these tasks recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the host to be restored.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).

2   To execute the Networker Administrative program at the command line prompt enter:

nwadmin &

- A window opens for the Networker Administrative program.
- You are now able to restore partitions.

3   Execute the following menu path:

Save Set → Recover . . . .

- The **Save Set Recover** window opens.

4   Execute the following menu path:

Clients → <system>

- **<system>** is the name of system to be restored (referred to as "system" in the rest of this procedure).
- The **Save Set** listing updates. This is a listing of partitions on the system.
- At this time, note the partitions listed for the system. To do a complete system restore, this procedure needs to be performed for each partition listed.

5   In the **Save Set** list **single-click** on the name of the partition for the restoration.
- The name is highlighted and the **Instances** listing is updated.

6   **Single-click** on the appropriate instance in the **Instances** listing.
  • An instance is a particular Networker client backup.
    − A listing of instances is a report detailed with the Networker client backups that
      have occurred.
  • Select an instance based upon the date from which to restore (referred to as "date" in
    the rest of this procedure) and an appropriate level.
    − To determine a base date, you must consider the time of day that backups occur.
      For example, if the backup occurs at 02:00 each morning then a system corrupted
      at noon on June 6 would require a restoration of the June 6 backup. However, if
      the system corruption took place around the time of the backup, it would be more
      prudent to use the backup from June 5.
  • If the backups are full or incremental, perform the following actions:
    − Select the most recent full backup that occurred on or prior to the date and
      perform a partition restore. If the date of this full backup is not the same as the
      date, perform a partition restore using each incremental backup, in chronological
      order, between this full backup and the day after the date.
  • If the backups are of different numerical levels, follow these steps:
    − First select the most recent level 0/full backup prior to or on the date and perform
      a restore of the partition. If a level 0/full backup did not occur on the date, select
      the most recent backup of the next highest level occurring after this level 0 and
      prior to or on the date.
    − Perform a restore of the partition.
    − Continue to select the most recent backup of the next highest level occurring
      between the last used instance and the day after the date until reaching an instance
      on the date.
  • You can double-click an instance to see which tape is required.

7   **Single-click** on the **Recover** button.
  • The **Save Set Recover Status** window opens.
  • **Single-clicking** on the **Volumes** button shows which tapes are required.

8   **Single-click** on the **Options** button.
  • The **Save Set Recover Options** window opens.

9   In the **Duplicate file resolution** area **single-click** on the **Overwrite existing file** radio
        button.

10   Ensure that the **Always prompt** checkbox is **not** checked (**single-click** on the checkbox
        if necessary).

11   **Single-click** on the **OK** button.
  • The **Save Set Recover Options** window closes.

12   **Single-click** on the **Start** button in the **Save Set Recover Status** window.
- Status messages appear in the **Status** box.
- If prompted for tapes, **single-click** the **Cancel** button in the **Save Set Recover Status** window and follow Steps 1 through 18 of the Index Tapes procedure (Section 3.3.6.2.2).
- A "recovery complete" message appears when recovery is complete.

13   **Single-click** on the **Cancel** button after the "recovery complete" message appears.
- The **Save Set Recover Status** window closes.

14   If additional partition restores are required, repeat Steps 6 through 13; otherwise go to Step 15.

15   To quit the Networker Administrative program when all desired restorations have been completed, execute the following menu path:

File → Exit

- The **nwadmin** window is closed.

16   At the UNIX prompt for the backup server enter:

exit

- The owner of the file(s) to be restored is logged out.

17   At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the machine to be restored.

Table 3.3-10. Perform Full System Restore - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the machine to be backed up** (Use procedure in Section 3.2.1)
2	**nwadmin &**	**enter text, press Enter**
3	**Save Set → Recover . . . .**	**single-click**
4	**Clients → <system>**	**single-click**
5	**<partition>** (to be restored) (in the **Save Set** list)	**single-click**
6	**<instance>** (in the **Instances** listing)	**single-click**
7	**Recover** button	**single-click**
8	**<date>** (from which to restore)	**single-click**
9	**Overwrite existing file** radio button (in the **Duplicate file resolution** area)	**single-click**
10	**Always prompt** checkbox **not** checked	**single-click** (if necessary)

Table 3.3-10. Perform Full System Restore - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
11	**OK** button (**Save Set Recover Options** window)	**single-click**
12	**Start** button (**Save Set Recover Status** window)	**single-click**
13	**Cancel** button (**Save Set Recover Status** window)	**single-click**
14	(No entry)	**Repeat Steps 6 through 13 as necessary to restore additional partitions**
15	**File → Exit** (**nwadmin** window)	**single-click**
16	**exit**	**enter text, press Enter**
17	**exit**	**enter text, press Enter**

3.3.6 Tape Handling

The following procedures describe how to label tapes, index tapes, and clean tape drives. Each of these procedures contains detailed steps that explain how to complete the procedure properly. Each tape handling procedure is significant in maintaining a working backup system. DAAC scheduled backups depend on proper maintenance of tape media and tape drives. Listed are complete explanations of the procedures and their relevance to the Computer Operator position.

The Activity Checklist in Table 3.3-11 provides an overview of the tape handling process.

Table 3.3-11. Tape Handling - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Labeling Tapes	(I) 3.3.6.1	
2	OPR	Indexing Tapes	(P) 3.3.6.2	
3	OPR	Tape Drive Cleaning	(P) 3.3.6.3	

3.3.6.1 Labeling Tapes

The Tape Labeling process begins when the Operator is performing procedure 3.3.1, Incremental Backup, or procedure 3.3.2, Full Backup, (or their associated Quick Steps) and runs out of tapes. The tape(s) must be installed in the jukebox and labeled. Networker uses tape labels for identification. The label that Networker creates is on the tape media itself, rather than a sticker on the outside of the tape cassette. An index is kept by Networker associating tape labels with particular backups/data. When you select files to be recovered using the Networker Recovery window or view saved sets on a backup volume using the Volume Management window in Networker, you are viewing this index. After labeling the required tape(s), the Operator resumes procedure 3.3.1 or 3.3.2 (as applicable).

Table 3.3-12. Labeling Tapes - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Install Required Tape(s) in Jukebox	(P) 3.3.6.1.1	
2	OPR	Perform Tape Labeling	(P) 3.3.6.1.2	

Detailed procedures for tasks performed by the Operator are provided in the sections that follow.

3.3.6.1.1   Install Required Tape(s) in Jukebox

The procedures assume that the Operator was previously executing procedure 3.3.1, Incremental Backup, or procedure 3.3.2, Full Backup. In order to perform the procedures, the Operator must have obtained the following:

- Blank tape(s)

All tapes are stored in the storage cabinet located in the control room. There are five tapes in each box, and every box of tapes has a unique number. To begin finding tapes for recycling to be labeled and installed in the Jukebox, the lowest numbers of a tape or a box of tapes should be used. Do not recycle any tape or box of tapes that the numbers are higher or current.

3.3.6.1.2   Perform Tape Labeling

Files and directories have unique names that are assigned by the user to identify them. In much the same manner, tapes are given unique names, or labels. This allows such programs as Networker and such hardware devices such as the Exabyte jukebox to automate the tape selection process when performing system backups and restores. When a tape is initialized, Networker assigns it a label. Networker then stores the tape's label with a file that is written to the tape so that when a file restoration request is received, Networker will know exactly which tape to select from the jukebox.

Table 3.3-13 presents the steps required to perform tape labeling. If you are already familiar with the procedure, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the Tape Backup Server.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).
- Examples of Tape Backup Server host names include **e0mss04, g0mss07**, **l0mss05**, and **n0mss05**.

2   To start logging in as root, at the command line prompt enter:

su

- A password prompt is displayed.

3   At the command line prompt enter:

<root password>

- Remember that the root password is case-sensitive.
- You are authenticated as root and returned to the UNIX prompt.

4   To execute the Networker Administrative program at the command line prompt enter:

nwadmin &

- The Networker Administrative program GUI is displayed.

5   Insert the blank tape(s) in the jukebox's cartridge.
- Remove all non-blank tapes from the cartridge or else they will be re-labeled and any data on the tapes will be lost.
- Slot 11 is the non-removable slot within the jukebox. This usually contains a cleaning tape. Do not enter any tape in Slot 11 for labeling.
- It is OK to leave empty slots.

6   Install the jukebox's cartridge in the jukebox.

7   **Single-click** on the **Label** button on the Networker Administrative program GUI.
- The **Jukebox Labeling** window opens.

8   In the **Starting with:** field enter:

<tape label>

- **<tape label>** is the label for the first tape in the sequence.
- Tape labels are named by using the host name (e.g., **x0acg01**), a dot or period, and a sequential number (e.g., **001**, **002**).
- By default, the system will prompt you with the next label in the sequence (e.g., **x0acg01.011**).
- After typing the text, press the **Tab** key to advance to the next field.
  - The cursor moves to the **First slot:** field.

9   In the **First slot:** field enter:

<number>

- **<number>** (e.g., **1**) refers to the slot containing the first volume to be labeled.
- Slot 1 is at the top of the cartridge.
- Slot 10 is near the bottom of the cartridge. (Slot 11 at the bottom of the cartridge normally holds the cleaning cartridge).
- After typing the text, press the **Tab** key to advance to the next field.
  - The cursor moves to the **Last slot:** field.

10   In the **Last slot:** field enter:

<number>

- **<number>** (e.g., **10**) refers to the slot containing the last volume to be labeled.
- Slot 10 is near the bottom of the cartridge. (Slot 11 at the bottom of the cartridge normally holds the cleaning cartridge).

11   **Single-click** on the **OK** button.
- A status message appears and updates.
- Labeling a full cartridge of tapes takes about 15 minutes.

12   When the status in the **Jukebox Labeling** window reads finished, **single-click** on the **Cancel** button.
- The **Jukebox Labeling** window closes.

13   To quit the Networker Administrative program GUI when labeling has been completed, execute the following menu path:

File → Exit

- The **nwadmin** program terminates and you are returned to the UNIX prompt.

14   At the UNIX prompt for the backup server enter:

exit

- **Root** is logged out.

15   At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the Tape Backup Server.

16   Attach an identifying sticker to the outside of each tape cassette.

Table 3.3-13. Perform Tape Labeling - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the Tape Backup Server** (Use procedure in Section 3.2.1)
2	**su**	**enter text, press Enter**
3	**<root password>**	**enter text, press Enter**
4	**nwadmin &**	**enter text, press Enter**
5	(No entry)	**insert the blank tape(s) in the jukebox's cartridge**
6	(No entry)	**install the jukebox's cartridge in the jukebox**

Table 3.3-13. Perform Tape Labeling - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
7	**Label** button	**single-click**
8	**<tape label>** (in the **Starting with:** field)	**enter text, press Tab**
9	**<number>** (in the **First slot:** field)	**enter text, press Tab**
10	**<number>** (in the **Last slot:** field)	**enter text**
11	**OK** button	**single-click**
12	**Cancel** button (when jukebox labeling is done)	**single-click**
13	**File → Exit** (**nwadmin** window)	**single-click**
14	exit	**enter text, press Enter**
15	exit	**enter text, press Enter**
16	(No entry)	**attach an identifying sticker to the outside of each tape cassette**

3.3.6.2 Indexing Tapes

The Indexing Tapes process begins when the Operator has finished the **Perform Tape Labeling** procedure (Section 3.3.6.1.2). If the tape(s) is/are not indexed/inventoried, Networker will not be aware of it/them. After indexing the required tape(s), the Operator resumes procedure 3.3.1, Incremental Backup, or procedure 3.3.2, Full Backup (as applicable).

The Activity Checklist in Table 3.3-14 provides an overview of the indexing tapes process.

Table 3.3-14. Indexing Tapes - Activity Checklist

Order	Role	Task	Section	Complete?
1	OPR	Pull Required Tape(s) from Tape Storage	(I) 3.3.5.2.1	
2	OPR	Index Tapes	(P) 3.3.5.2.2	

3.3.6.2.1 Pull Required Tape(s) from Tape Storage

In order to perform the procedure, the Operator must have obtained the following:

- The required tape(s).

This may necessitate retrieving tapes from secure offsite storage if other backups are unavailable.

3.3.6.2.2 Index Tapes

Detailed procedures for tasks performed by the Operator are provided in the sections that follow.

The procedures assume that the Operator has previously executed procedure 3.3.6.1, Tape Labeling.

Table 3.3-15 presents the steps required to index tapes. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the Tape Backup Server.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).
- Examples of Tape Backup Server host names include **e0mss04, g0mss07**, **l0mss05**, and **n0mss05**.

2   To start logging in as root, at the command line prompt enter:

su

- A password prompt is displayed.

3   At the command line prompt enter:

<root password>

- Remember that the root password is case-sensitive.
- You are authenticated as root and returned to the UNIX prompt.

4   To execute the Networker Administrative program at the command line prompt enter:

nwadmin &

- The Networker Administrative program GUI is displayed.
- You are now able to index tapes.

5   **Single-click** on the **Mount** button.
- The **Jukebox Mounting** window is displayed.

6   Observe the entries in the **Jukebox Mounting** window to see what tapes are currently available to Networker.

7   **Single-click** on the **Cancel** button to dismiss the **Jukebox Mounting** window.
- The **Jukebox Mounting** window is closed.

8   Insert the required tape(s) in the jukebox's cartridge.
- For instructions, refer to the jukebox's documentation.

9   Install the jukebox's cartridge in the jukebox.

10   Execute the following menu path:

       Media → Inventory

- The **Jukebox Inventory** window opens.

11   In the **First slot:** field enter:

       <number>

- **<number>** (e.g., **1**) refers to the slot containing the first volume to be indexed.
- Slot 1 is at the top of the cartridge.
- Slot 10 is near the bottom of the cartridge. (Slot 11 at the bottom of the cartridge normally holds the cleaning cartridge).
- After typing the text, press the **Tab** key to advance to the next field.
  - The cursor moves to the **Last slot:** field.
- It is OK to have empty slots or slots with tapes that have already been indexed.

12   In the **Last slot:** field enter:

       <number>

- **<number>** (e.g., **10**) refers to the slot containing the last volume to be indexed.
- Slot 10 is near the bottom of the cartridge. (Slot 11 at the bottom of the cartridge normally holds the cleaning cartridge).

13   **Single-click** on the **OK** button.
- A checking volume message appears and updates.
- Performing an inventory on a full cartridge takes twenty to thirty minutes.

14   When the status in the **Jukebox Inventory** window reads finished, **single-click** on the **Cancel** button.
- The **Jukebox Inventory** window closes.

15   **Single-click** on the **Mount** button.
- The **Jukebox Mounting** window is displayed.

16   Observe the entries in the **Jukebox Mounting** window to determine whether the indexing was successful (if the required tape(s) is/are displayed in the window).

17   **Single-click** on the **Cancel** button to dismiss the **Jukebox Mounting** window.
- The **Jukebox Mounting** window is closed.

18   If the required tape(s) were not displayed in the **Jukebox Mounting** window, go back to Step 10.

19   If the required tape(s) were displayed in the **Jukebox Mounting** window, execute the following menu path:

File → Exit

- The **nwadmin** program terminates and you are returned to the UNIX prompt.

20   At the UNIX prompt for the backup server enter:

exit

- **Root** is logged out.

21   At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the Tape Backup Server.

Table 3.3-15. Index Tapes - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the Tape Backup Server** (Use procedure in Section 3.2.1)
2	**su**	**enter text, press Enter**
3	**<root password>**	**enter text, press Enter**
4	**nwadmin &**	**enter text, press Enter**
5	**Mount** button	**single-click**
6	(No entry)	**determine what tapes are currently available to Networker (Jukebox Mounting window)**
7	**Cancel** button (**Jukebox Mounting** window)	**single-click**
8	(No entry)	**insert the required tape(s) in the jukebox's cartridge**
9	(No entry)	**install the jukebox's cartridge in the jukebox**
10	**Media → Inventory**	**single-click**
11	**<number>** (in the **First slot:** field)	**enter text, press Tab**
12	**<number>** (in the **Last slot:** field)	**enter text**
13	**OK** button	**single-click**
14	**Cancel** button (when jukebox indexing is done)	**single-click**
15	**Mount** button	**single-click**
16	(No entry)	**determine whether the indexing was successful (if the required tape(s) is/are displayed in the Jukebox Mounting window)**
17	**Cancel** button (**Jukebox Mounting** window)	**single-click**

Table 3.3-15. Index Tapes - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
18	(No entry)	**if the required tape(s) were not displayed in the Jukebox Mounting window, go back to Step 10**
19	**File → Exit** (**nwadmin** window)	**single-click**
20	**exit**	**enter text, press Enter**
21	**exit**	**enter text, press Enter**

3.3.6.3 Perform Tape Drive Cleaning

The system will at times prompt for drive(s) cleaning, typically during non-processing periods. However, during the course of the tape backup process period, the drive(s) may send a request for cleaning. Manual cleaning should be performed each time tapes are installed in the jukebox. Maintaining clean drives can help prevent backup interruption that may occur due to unclean tape drive heads.

Table 3.3-16 presents the steps required to perform tape drive cleaning. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Access a terminal window logged in to the Tape Backup Server.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 3.1.4).
- Examples of Tape Backup Server host names include **e0mss04, g0mss07**, **l0mss05**, and **n0mss05**.

2   To start logging in as root, at the command line prompt enter:

su

- A password prompt is displayed.

3   At the command line prompt enter:

<root password>

- Remember that the root password is case-sensitive.
- You are authenticated as root and returned to the UNIX prompt.

4   To execute the Networker Administrative program at the command line prompt enter:

nwadmin &

- The Networker Administrative program GUI is displayed.

5   In the **Devices:** field **single-click** on (highlight) a drive that the system has prompted for cleaning.

6   **Single-click** on the **Unmount** button.

7   Wait a few minutes for the drive to unmount completely.

8   Repeat Steps 5 through 7 to unmount the second drive in the jukebox (if applicable).

9   To open the jukebox door turn the key counterclockwise in the lock.

10   Remove the last tape from the bottom of the cartridge.
- The last tape from the bottom of the cartridge is the cleaning tape.

11   Observe the cleaning tape to determine whether it is still usable.
- Cleaning tapes expire after several uses. After each use the appropriate box on the surface of the tape is marked to maintain a list of usage.

12   Insert the cleaning tape gently into one of the drives in the jukebox.

13   Wait until the tape has been ejected and the flashing lights on the drive are off before removing the tape from the drive.

14   Remove the tape from the drive when cleaning has finished.

15   Mark the appropriate box on the surface of the tape to maintain a list of usage.

16   Repeat Steps 11 through 15 to clean the other drive in the jukebox.

17   Insert the cleaning tape in the last slot in the jukebox's cartridge.

18   Install the jukebox's cartridge in the jukebox.

19   Close and lock the jukebox door (turn the key clockwise in the lock).

20   If appropriate, return to the task that was interrupted by a prompt to clean the tape drive(s).

21   If there is no other task requiring the use of the Networker Administrative program GUI, execute the following menu path:

File → Exit

- The **nwadmin** program terminates and you are returned to the UNIX prompt.

22   At the UNIX prompt for the backup server enter:

exit

- **Root** is logged out.

23   At the UNIX prompt re-enter:

exit

- You are logged out and disconnected from the Tape Backup Server.

Table 3.3-16. Perform Tape Drive Cleaning - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	(No entry)	**access a terminal window logged in to the Tape Backup Server** (Use procedure in Section 3.2.1)
2	**su**	**enter text, press Enter**
3	**<root password>**	**enter text, press Enter**
4	**nwadmin &**	**enter text, press Enter**
5	**<drive>** (to be cleaned) (in the **Devices:** field)	**single-click**
6	**Unmount** button	**single-click**
7	(No entry)	**wait a few minutes for the drive to unmount completely**
8	(No entry)	**repeat Steps 5 through 7 to unmount the second drive in the jukebox (if applicable)**
9	(No entry)	**open the jukebox door (turn the key counterclockwise in the lock)**
10	(No entry)	**remove the last tape from the bottom of the cartridge**
11	(No entry)	**observe the cleaning tape to determine whether it is still usable**
12	(No entry)	**insert the cleaning tape gently into one of the drives in the jukebox**
13	(No entry)	**wait until the tape has been ejected and the flashing lights on the drive are off**
14	(No entry)	**remove the tape from the drive**
15	(No entry)	**mark the appropriate box on the surface of the tape to maintain a list of usage**
16	(No entry)	**repeat Steps 11 through 15 to clean the other drive in the jukebox**
17	(No entry)	**insert the cleaning tape in the last slot in the jukebox's cartridge**
18	(No entry)	**install the jukebox's cartridge in the jukebox**
19	(No entry)	**close and lock the jukebox door (turn the key clockwise in the lock)**

Table 3.3-16. Perform Tape Drive Cleaning - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
20	(No entry)	**return to the task that was interrupted by a prompt to clean the tape drive(s) (as applicable)**
21	**File → Exit** (**nwadmin** window) (if applicable)	**single-click**
22	**exit** (if applicable)	**enter text, press Enter**
23	**exit** (if applicable)	**enter text, press Enter**

3.4 User Administration

User Administration procedures will be implemented through command-line and/or script entries.

3.4.1 Adding a User

The Adding a User process begins when the requester fills out a "User Registration Request Form" and submits it to the site supervisor. The "User Registration Request Form" includes information regarding the user (User's Name, Group, Organization, etc.), as well as the user's explanation of why an account on the system is needed. The requester's supervisor reviews the request, and if it is determined that it is appropriate for the requester to have a UNIX account, forwards the request to the Operations Supervisor (OPS Super). The OPS Supervisor reviews the request and forwards it to the System Administrator (SA). The SA verifies that all required information is contained on the form. If it is, the SA implements the request. (Incomplete forms are returned to the requester's supervisor for additional information.)   After the user is registered, the SA provides the user with a password to use for logging onto their accounts. The SA also sends an e-mail message to the user's supervisor and the OPS Super, informing them that the user's accounts were created.

Table 3.4-1, the Activity Checklist table that follows, provides an overview of the adding a user process.

Table 3.4-1. Adding a User - Activity Checklist (1 of 2)

Order	Role	Task	Section	Complete?
1	Requester	Complete User Registration Form and forward to the Supervisor	(I) 3.4.1	
2	Supervisor	Approve/Deny Request. If Approve, Forward Request to OPS Super	(I) 3.4.1	
3	OPS Supervisor	Review Request and Forward to SA	(I) 3.4.1	
4	SA	Review User Registration Form for Completeness	(I) 3.4.1	
5	SA	Add User	(P) 3.4.1	

Table 3.4-1. Adding a User - Activity Checklist (2 of 2)

Order	Role	Task	Section	Complete?
6	SA	Phone/e-mail User with Password. Notify Supervisor and OPS Supervisor that user was added	(I) 3.4.1	

Depending upon the script utilized, in order to add a new user the SA should obtain information such as the following about the requester:

- Real name of the new user
- User name of the new user
- Office number of the new user
- Office phone number of the new user
- Home phone number of the new user
- Organization
- Group affiliation(s)
- Role(s) of the new user

The SA creates a new user account with command-line/script entries. As an example, The Goddard Space Flight Center DAAC uses a script, Newuser, to add new users to the system. The script, which is available to other DAACs, walks the System Administrator through data input of user information, checks for the same user in other systems, creates a User ID, synchronizes password files and creates home directories for new users.

3.4.2 Deleting a User

The Deleting a User process begins when the requester has determined that no useful files remain in the user's home directory and submits a request to delete the user's account to the user's supervisor. The supervisor approves or denies the request. Once approved, the request is forwarded to the OPS Supervisor. The OPS Supervisor reviews the request and forwards it to the SA, who deletes the user's account. When the user has been deleted, the SA notifies the requester, supervisor and OPS Super.

The Activity Checklist table that follows provides an overview of the deleting a user account process.

Table 3.4-2. Deleting a User - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Determine that No Useful Files Remain in the User's Home Directory and Submit Request to user's Supervisor	(I) 3.4.2	
2	OPS Supervisor	Approve/Deny Request. If Approve, Forward Request to OPS Super.	(I) 3.4.2	
3	OPS Supervisor	Review Request and Forward to SA	(I) 3.4.1	
4	SA	Delete User	(P) 3.4.2	
5	SA	Add User	(I) 3.4.2	

The process assumes that the requester's application for deleting a user has already been approved by DAAC Management. In order to perform the procedure, the SA must have obtained the following information from the requester:

- UNIX login of the user to be deleted
- Role(s) of the user to be deleted

The SA deletes a user with command-line/script entries. As an example, The Goddard Space Flight Center DAAC uses a script, Lockdown, to lock, unlock and delete user accounts. This script, which is available to other DAACs, walks the System Administrator through the steps necessary to delete a user account. It assists the System Administrator in locating the correct user account for deletion, deletes the user account and all associated file references. It also enables the System Administrator to lock or unlock accounts.

3.4.3 Changing a User Account Configuration

The Changing a User Account Configuration process begins when the requester submits a request to the OPS Supervisor detailing what to change about the account configuration and the reason for the change. The OPS Supervisor reviews the request and forwards it to SA who changes the user's account configuration. When the changes are complete the SA notifies the requester and OPS Supervisor.

Table 3.4-3, the Activity Checklist table that follows, provides an overview of the changing a user account configuration process.

Table 3.4-3. Change a User Account Configuration - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request to OPS Supervisor	(I) 3.4.3	
2	OPS Supervisor	Review and Forward to SA	(I) 3.4.3	
3	SA	Change User Account Configuration	(P) 3.4.3	
4	SA	Inform Requester and Supervisor of completion	(I) 3.4.3	

The process assumes that the requester's application for changing a user account configuration has already been approved by the OPS Supervisor. In order to perform the procedure, the SA must have obtained the following information from the requester:

- What to change and new settings. Can be any of:
  - New Real User Name
  - New Login ID
  - New Office Number
  - New Office Phone Number
  - New Home Phone Number
  - New UNIX Group
  - New Login Shell
- Current UNIX Login of the User

The SA changes the appropriate configuration items manually in the users home directory.

3.4.4 Changing User Access Privileges

The Changing User Access Privileges process begins when the requester submits a request to the supervisor. The supervisor approves or denies the request. Once approved, the request is forwarded to the OPS Supervisor. The OPS Supervisor reviews the request and forwards it to the SA who changes the user's access privileges. When the changes are complete the SA notifies the requester, supervisor and OPS Super.

Table 3.4.4, the Activity Checklist table that follows, provides an overview of the changing user access privileges process.

Table 3.4-4. Changing User Access Privileges - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request to the Supervisor	(I) 3.4.4	
2	Supervisor	Approve/Deny Request. If Approve, Forward Request to OPS Super	(I) 3.4.4	
3	OPS Supervisor	Review Request and Forward to SA	(I) 3.4.4	
4	SA	Change User Access Privileges	(P) 3.4.4	
5	SA	Inform Requester, Supervisor and DAAC Mgr of completion	(I) 3.4.4	

Detailed procedures for tasks performed by the SA are provided in the sections that follow.

The procedures assume that the requester's application for changing user access privileges has already been approved by DAAC Management and that the SA is an administrator. In order to perform the procedure, the SA must have obtained the following information about the requester:

- Role(s) to which the user is to be added
- Role(s) from which the user is to be removed

- UNIX login of the user

To change user access privileges for the requester, execute the procedure steps that follow:

3.4.5 Changing a User Password

The Changing a User Password process begins when the requester submits a request to the SA. The SA verifies that the requester is who s/he claims to be. Once verified, the SA changes the user's password. When the change is complete the SA notifies the requester.

Table 3.4-5, the Activity Checklist table that follows, provides an overview of the changing a user password process.

Table 3.4-5. Changing a User Password - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request to SA	(I) 3.4.5	
2	SA	Verify that the Requester is Who S/he Claims to Be	(I) 3.4.5	
3	SA	Change Password	(P) 3.4.5	
4	SA	Inform Requester of completion	(I) 3.4.5	

Detailed procedures for tasks performed by the SA are provided in the sections that follow.

The procedures assume that the requester's application for changing a user password has already been approved by DAAC Management. In order to perform the procedure, the SA must have obtained the following information about the requester:

- UNIX login of the user
- New password for the user

To change a user password for the requester, execute the procedure steps that follow:

3.4.6 Checking a File/Directory Access Privilege Status

The Checking a File/Directory Access Privilege Status process begins when the requester submits a request to the SA. The SA checks the file/directory access privilege status and reports the status back to the requester.

Table 3.4-6, the Activity Checklist that follows provides an overview of the checking a file/directory access privilege status process.

***Table 3.4-6. Checking the Access Privilege Status of a File/Directory -
Activity Checklist***

Order	Role	Task	Section	Complete?
1	Requester	Submit a Request to the SA	(I) 3.4.6	
2	SA	Check the Access Privilege Status of a File/Directory	(P) 3.4.6.1	

Detailed procedures for tasks performed by the SA are provided in the sections that follow. In order to perform the procedure, the SA must have obtained the following information about the requester:

- Full path of the file/directory on which privilege status is needed.

3.4.6.1 Check the Access Privilege Status of a File/Directory

Table 3.4-7 contains a table that presents the steps required to check a file/directory access privilege status in a condensed manner. If you are already familiar with the procedure, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedure presented below.

To check a file/directory access privilege status for the requester, execute the procedure steps that follow:

1   At the UNIX command line prompt enter:

cd <path>

- The path is the full path up to but not including the file/directory on which access privilege status is needed.
- For example, if the requester wants access privilege status on directory /home/jdoe then enter:

    cd /home

2   At the UNIX command line prompt enter:

ls -la | grep <FileOrDirectoryName>

- The command returns the following type of information:

    drwxr-xr-x   19 jdoe user   4096 Jun 28 09:51 jdoe
    -r-xr--r--   1   jdoe user   80   Jun 22 11:22 junk

- What this output means, from left to right, is:
    - The file type and access permissions:

        The first character indicates what type of file it is:
        d means that the file is a directory.
        - means that the file is an ordinary file.
        l means that the file is a symbolic link.

- The next three characters indicate the owner privileges, in the order: r = read   w = write   x = execute.   - is a placeholder. Example: the owner (jdoe) of the file junk does not have write permissions, so a - appears rather than a w.
- The next three characters indicate the group privileges, in the order: r = read   w = write   x = execute.   - is a placeholder. Example: the group (user) of the directory jdoe does not have write permissions, so a - appears rather than a w as the sixth character in the line.
- The next three characters indicate the privileges that everyone else/other has, in the order: r = read   w = write   x = execute.   - is a placeholder. Example: other in the case of the directory jdoe does not have write permissions, so a - appears rather than a w as the ninth character in the line.

There are 19 links to the file/directory jdoe.
The owner of the file/directory is jdoe.
The file/directory's group is user.
The file/directory is 4096 bytes large.
The last time the file/directory was modified is Jun 28 at 09:51.
The name of the file/directory is jdoe.

3   To create a report of the access privilege status of the file/directory in the template in Figure 3.4-1 enter:

<text>

4   Report the file/directory access privilege status to the requester.

```
full path of the file/directory: _____

owner: _____

group: _____

owner/user privileges:          _____ read _____ write _____ execute

group privileges:               _____ read _____ write _____ execute

everyone else/other privileges: _____ read _____ write _____ execute
```

Figure 3.4-1. Template for Reports of the Access Privilege Status of a File/Directory

Table 3.4-7. Check the Access Privilege Status of a File/Directory - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	cd <path>	enter text, press Enter
2	ls -la | grep <FileOrDirectoryName>	enter text, press Enter
3	<text>	enter text, press Enter
4	(No entry)	report the file/directory access privilege status to the requester

3.4.7 Changing the Access Privilege of a File/Directory

The process of changing the access privilege of a file/directory begins when the requester submits a request to the supervisor to have file/directory access privileges changed. The supervisor approves/denies the request. When approved, the request is forwarded to the OPS Supervisor who reviews the request and forwards it to the SA. The SA changes the file/directory access privileges and then notifies the requester, supervisor and OPS Supervisor of completion.

Table 3.4-8, the Activity Checklist table that follows, provides an overview of the process of changing the access privilege of a file/directory.

Table 3.4-8. Changing the Access Privilege of a File/Directory - Activity Checklist

Order	Role	Task	Section	Complete?
1	Requester	Submit Request to the Supervisor	(I) 3.4.7	
2	Supervisor	Approve/Deny Request. If Approve, Forward Request to OPS Supervisor	(I) 3.4.7	
3	OPS Supervisor	Review Request and Forward to SA	(I) 3.4.7	
4	SA	Change the Access Privilege of a File/Directory	(P) 3.4.7	

Detailed procedures for tasks performed by the SA are provided in the sections that follow.

The procedures assume that the requester's application for changing a file/directory access privilege has already been approved by the supervisor. In order to perform the procedure, the SA must have obtained the following information about the requester:

- Full path of the file/directory on which access privileges will be changed.
- New access privileges to set on the file/directory. Can be any of:
    - New owner.
    - New group.
    - New user/owner privileges (read, write and/or execute).
    - New group privileges (read, write and/or execute).
    - New other privileges (read, write and/or execute).

Table 3.4-9 contains a table that presents the steps required to change the access privilege of a file/directory in a condensed manner. If you are already familiar with the procedure, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedure presented below.

1   At the UNIX command line prompt enter:

su

2   At the command line prompt enter:

<RootPassword>

- Remember that **<RootPassword>** is case-sensitive.
- You are authenticated as root.

3   At the command line prompt enter:

cd <path>

- The path is the full path up to but not including the file/directory on which access privileges will be changed.

- For example, if the requester wants access privileges changed on directory /home/jdoe enter:

  cd /home

4   To assign a file/directory to a new owner at the command line prompt enter:

chown \<NewOwner> \<FileOrDirectoryName>

- The FileOrDirectoryName is the name of the file/directory on which access privileges will be changed minus the path.
- For example, if the requester wants to assign access privileges on directory **/home/jdoe** to new owner **johnd** enter:

  chown johnd jdoe

5   To assign a file/directory to a new group at the command line prompt enter:

chgrp \<NewGroup> \<FileOrDirectoryName>

- The FileOrDirectoryName is the name of the file/directory on which access privileges will be changed minus the path.
- For example, if the requester wants to assign directory **/home/jdoe** to group **emd** enter:

  chgrp emd jdoe

6   To assign new user/owner privileges to a file/directory at the command line prompt enter:

chmod u=\<NewUserPrivileges> \<FileOrDirectoryName>

- The NewUserPrivileges are as follows:
  - r = read.
  - w = write.
  - x = execute.
- For example, to give the user/owner **read**, **write** and **execute** privileges on directory **/home/jdoe**, enter:

  chmod u=rwx /home/jdoe

7   To assign new group privileges at the command line prompt enter:
chmod g=\<NewGroupPrivileges> \<FileOrDirectoryName>
- For example, to give the group **read** and **execute** privileges on the directory **/home/jdoe** enter:

  chmod g=rx /home/jdoe

8   To assign new privileges to "other" users at the command line prompt enter:

chmod o=<NewOtherPrivileges> <FileOrDirectoryName>

- For example, to give **read** privileges on the directory **/home/jdoe** to "other" users enter:

chmod o=r /home/jdoe

9   To exit, at the command line prompt enter:

exit

- Root is logged out.

Table 3.4-9. Change the Access Privilege of a File/Directory -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**su**	**enter text, press Enter**
2	**<RootPassword>**	**enter text, press Enter**
3	**cd <path>**	**enter text, press Enter**
4	**chown <NewOwner> <FileOrDirectoryName>** (if applicable)	**enter text, press Enter r**
5	**chgrp <NewGroup> <FileOrDirectoryName>** (if applicable)	**enter text, press Enter**
6	**chmod u=<NewUserPrivileges> <FileOrDirectoryName>** (if applicable)	**enter text, press Enter**
7	**chmod g=<NewGroupPrivileges> <FileOrDirectoryName>** (if applicable)	**enter text, press Enter**
8	**chmod o=<NewOtherPrivileges> <FileOrDirectoryName>** (if applicable)	**enter text, press Enter**
9	**exit** (when applicable)	**enter text, press Enter**
10	**(No entry)**	**inform requester, supervisor and OPS Supervisor of completion**

3.5 Security

EMD security architecture must meet the requirements for data integrity, availability, and confidentiality. EMD Security Services meets these requirements by incorporating a variety of mechanisms to establish and verify user accounts, issue and verify passwords, audit user activity, and verify and protect data transfer. Security logs are monitored and security reports generated by the System Administrator as required.   Several open source products provide tools for authentication and network and systems monitoring - Crack, ANLpasswd, TCP Wrappers, and Tripwire.   Crack and ANLpasswd provide brute force password cracking and password checking, respectively for local system and network access. Tripwire monitors for intruders by noting changes to files. F-Secure Secure Shell (ssh) provides strong authentication access and

session encryption from external, non-trusted networks as well as internally within a DAAC. Security Services also supports detection of, reporting, and recovery from security breaches.

Chapter 5 defines step-by-step procedures for Operations personnel to run the Security Services tools. This section describes procedures for generating security reports.

3.5.1 Generating Security Reports

3.5.1.1 Review User Activity Data

A log is created to keep track of unsuccessful attempts to log into the computer. After a person makes five consecutive unsuccessful attempts to log in, all these attempts are recorded in the file /var/adm/loginlog. The procedures assume that the file has been created and the operator has logged on as root.

Table 3.5-1 contains a table that presents the steps required to review user activity data in a condensed manner. If you are already familiar with the procedure, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedure presented below.

1   At the UNIX command line prompt enter:

   /usr/bin/logins [-admopstux] [-g group..] [-l login..]

2   At the command line prompt enter:

   logins –x   -l <username>

   • Displays login status for a user:

3   At the command line prompt enter:

   /var/adm/loginlog

   • To enable login logging, this creates the log file loginlog.

4   At the command line prompt enter:

   chmod 600 /var/adm/loginlog

   • This sets read and write permissions for root on the file.

5   At the command line prompt enter:

   chgrep sys /var/adm/loginlog

   • This sets the group to sys.

Table 3.5-1. Review User Activity Data - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	/usr/bin/logins [-admopstux] [-g group..] [-l login..]	enter text, press Enter
2	logins –x -l <username>	enter text, press Enter
3	/var/adm/loginlog	enter text, press Enter
4	chmod 600 /var/adm/loginlog	enter text, press Enter r
5	chgrep sys /var/adm/loginlog	enter text, press Enter

3.5.1.2 Monitor and Review User Audit Trail Information

The audit_startup script is used to initialize the audit subsystem before the audit daemon is started. This script is configurable by the System Administrator, and currently consists of a series of auditconfig commands to set the system default policy, and to download the initial events to class mapping. Type the following command to initialize the audit subsystem:

/etc/security/audit_startup

The audit command is the general administrator's interface to the audit trail. The audit daemon may be notified to read the contents of the audit_control file and re-initialize the current audit directory to the first directory listed in the audit_control file or to open a new audit file in the current audit directory specified in the audit_control file as last read by the audit daemon. The audit daemon may also be signaled to close the audit trail and disable auditing. The audit commands are input as shown:

Table 3.5-2 contains a table that presents the steps required to monitor and review user audit trail information in a condensed manner. If you are already familiar with the procedure, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedure presented below.

1   At the UNIX command line prompt enter:

audit -n

- Signals audit daemon to close the current audit file and open a new audit file in the current audit directory.

2   At the command line prompt enter:

audit -s

- Signals audit daemon to read the current audit file.
- The audit daemon stores the information internally.

3   At the command line prompt enter:

audit -t

- Signals audit daemon to close the current audit file, disable audit and die.

4   At the command line prompt enter:

praudit -sl <filename>

- Displays audit output.
- The print audit command converts the binary audit records into a variety of formats, depending on the options used with the commands.
- The format of audit files is included in the file /usr/include/sys/audit.h.
- By default, user IDs (UID) and group IDs (GID) are converted to their ACSII representation.

Table 3.5-2. Monitor and Review User Audit Trail Information -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**audit -n**	**enter text, press Enter**
2	**audit -s**	**enter text, press Enter**
3	**audit -t**	**enter text, press Enter**
4	**praudit -sl <filename>**	**enter text, press Enter r**

This page intentionally left blank.

4. Database Administration

Basic Database Administrator (DBA) tasks and procedures are described in the sections that follow. Table 4-1 shows DBA tasks that have to be done on a regular basis and the section where they are addressed in this document.

Table 4-1. Database Administrator (DBA) Tasks Performed on a Regular Basis

Time Period	Task	Importance	Found In …
Daily	Capture database configurations	Absolutely necessary for database recovery if problems occur	Configuring Databases
Weekly	Monitor Sybase disk usage		Monitoring and Tuning Databases
	Clean up old files		
Monthly	Reboot		Starting and Stopping Servers
Before and After Installations	Run DbVerify scripts		Installing Databases and Patches

Subsequent sections related to Database Administration address the following topics:

- Section 4.1 An overview of the process and step-by-step procedures for starting and stopping database servers.
- Section 4.2 An overview of the process and step-by-step procedures for creating database devices and logical volumes.
- Section 4.3 An overview of the process and step-by-step procedures for installing databases and patches.
- Section 4.4 An overview of the process and step-by-step procedures for configuring databases.
- Section 4.5 An overview of the process and step-by-step procedures for working with indexes, segments, and caches.
- Section 4.6 An overview of the process and step-by-step procedures for backing up and recovering data.
- Section 4.7 An overview of the process and step-by-step procedures for establishing database security.
- Section 4.8 An overview of the process and step-by-step procedures for copying individual databases.
- Section 4.9 An overview of the process and step-by-step procedures for bulk copying.

- Section 4.10   An overview of the process and step-by-step procedures for replication system administration.
- Section 4.11   An overview of the process and step-by-step procedures for performance monitoring, tuning, and problem reporting.
- Section 4.12   An overview of the process and step-by-step procedures for ensuring database quality.
- Section 4.13   An overview of the process and step-by-step procedures for Sybase troubleshooting.
- Section 4.14   An overview of the process and step-by-step procedures for performing Oracle procedures.

4.1 Starting and Stopping Database Servers

Each procedure outlined has an **Activity Checklist** table that provides an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one - *Order* shows the order in which tasks could be accomplished.

Column two - *Role* lists the Role/Manager/Operator responsible for performing the task.

Column three -*Task* provides a brief explanation of the task.

Column four - *Section* provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - **Complete?** is used as a checklist to keep track of which task steps have been completed.

Table 4.1-1, below, provides an Activity Checklist for starting and stopping database servers.

Table 4.1-1. Starting and Stopping Database Servers - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Start ASE Servers	(P) 4.1.1	
2	DBA	Start the ASE Backup Server	(P) 4.1.2	
3	DBA	Start the ASE Monitor Server	(P) 4.1.3	
4	DBA	Stop the ASE Monitor Server	(P) 4.1.4	
5	DBA	Stop the ASE Backup Server	(P) 4.1.5	
6	DBA	Stop the ASE Server	(P) 4.1.6	

4.1.1 Start Adaptive Server Enterprise (ASE) Servers

A server process is started manually when a new server is installed or after a system outage. In order to perform the procedure, the DBA must know the database server name and be assigned **sa_role** (refer to Section 4.7, Establishing Database Security). The following procedures are applicable.

1 Log in to a local host.

2 Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press the **Return/Enter** key.

3 Log into the server on which the ASE Server Process is to be started by typing: **/tools/bin/ssh** *ServerName*, then press the **Return/Enter** key.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4 If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, type your *Passphrase* and then press the **Return/Enter** key.
- Go to step 6.

5 At the <*user@remotehost*>**'s password:** prompt, type your *Password* and then press the **Return/Enter** key.

6 Log into the server as sybase, type, *su – sybase* and press the **Return/Enter** key.
- A password prompt is displayed.

7 Enter *sybase password*, then press the **Return/Enter** key.
- You are authenticated as yourself and returned to the UNIX prompt.

8 At the UNIX prompt, type **cd /ASE-12_5/install** and then press the **Return/Enter** key.

9 At the UNIX prompt, type **./RUN_***servername* **&** then press the **Return/Enter** key.

10 At the UNIX prompt, type **showserver**, then press the **Return/Enter** key.
- This displays a listing of the ASE Server processes that are running.

4.1.2 Start the ASE Backup Server

NOTE: This procedure should be done after ASE Server is up and running

1 Log in to a local host.

2 Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press the **Return/Enter** key.

3   Log into the server on which the ASE Server Process is to be started by typing:
/tools/bin/ssh *ServerName*, then press the **Return/Enter** key.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, type your *Passphrase* and then press the **Return/Enter** key.
- Go to step 6.

5   At the <*user@remotehost*>**'s password:** prompt, type your *Password* and then press the **Return/Enter** key.

6   Log into the server as sybase, type, *su – sybase* and press the **Return/Enter** key.
- A password prompt is displayed.

7   Enter *sybase password*, then press the **Return/Enter** key.
- You are authenticated as yourself and returned to the UNIX prompt.

8   Type **cd /ASE-12_5/install** and press the **Return/Enter** key.

9   Type, **./RUN_***backupservername* **&** and press the **Return/Enter** key.

NOTE: You may have to hit the **Return/Enter** key one more time

4.1.3 Start the ASE Monitor Server

NOTE: This procedure is be done on an as needed basis and after the ASE Server is up and running

1   Log in to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press the **Return/Enter** key.

3   Log into the server on which the ASE Server Process is to be started by typing:
/tools/bin/ssh *ServerName*, then press the **Return/Enter** key.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4 If a prompt to **Enter passphrase for RSA key '*<user@localhost>*'** appears, type your *Passphrase* and then press the **Return/Enter** key. Go to step 6.

5 At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the **Return/Enter** key.

6 Log into the server as sybase, type, *su – sybase* and press the **Return/Enter** key.
  - A password prompt is displayed.

7 Enter *sybase password*, then press the **Return/Enter** key.
  - You are authenticated as yourself and returned to the UNIX prompt.

8 Type **cd /ASE-12_5/install** and press the **Return/Enter** key.

9 Type **startserver –f run_monitor_srvr** and press the **Return/Enter** key.

4.1.4 Stop the ASE Monitor Server

Servers are stopped by the DBA when the system needs to be brought down for maintenance. Servers are brought down in an order that is the reverse of the start-up order. The **shutdown** command issued from within isql shuts down the server where you are logged in. It allows for the completion of any current processes and blocks the start of any new processes before the server is shutdown.

When you issue a **shutdown** command, the server:
  - Disables all logins except the sa's.
  - Performs a checkpoint in each database, moving changed pages from memory to disk.
  - Waits for currently executing SQL statements or procedures to finish.

Adding a **nowait** option to the command immediately terminates all processes and shuts down the server. The **nowait** option should be used sparingly because it may cause data loss and complicate recovery.

NOTE: This procedure should be done before the ASE Server is shutdown

1 Log on to a local host.

2 Set display to current terminal by typing: **setenv DISPLAY *IPNumber*:0.0** or **setenv DISPLAY *ServerName*:0.0**, then press the **Return/Enter** key.

3   Log into the server on which the ASE Server Process is to be stopped by typing:
/tools/bin/ssh *ServerName*, then press the **Return/Enter** key.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a
prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears; continue
with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears, type your
Passphrase and then press the **Return/Enter** key.
- Go to step 6.

5   At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the
Return/Enter key.

6   Log into the server as sybase, type, *su – sybase* and press the **Return/Enter** key.
- A password prompt is displayed.

7   Enter *sybase password*, then press the **Return/Enter** key.
- You are authenticated as yourself and returned to the UNIX prompt.
- Your new home directory is **/usr/ecs/OPS/COTS/sybase** and all required
environment variables have been set.

NOTE: Environment and DS queary are set when you su -sybase

8   Type **$SYBASE/bin/isql –Usa** and press the **Return/Enter** key.

9   Type the *sa-password* when prompted for it and press the **Return/Enter** key.

10   Type **sms_shutdown** at "**1>**" prompt and press the **Return/Enter** key.

11   Type **go** at "**2>**" prompt and press the **Return/Enter** key.

4.1.5 Stop the ASE Backup Server

NOTE: This procedure should be done before shutting down the ASE Server

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv
DISPLAY** *ServerName***:0.0**, then press the **Return/Enter** key.

3 Log into the server on which the ASE Server Process is to be stopped by typing: **/tools/bin/ssh** *ServerName*, then press the **Return/Enter** key.

- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4 If a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears, type your *Passphrase* and then press the **Return/Enter** key.

- Go to step 6.

5 At the **<*user@remotehost*>'s password:** prompt, type your *Password* and then press the **Return/Enter** key.

6 Log into the server as sybase, type, *su – sybase* and press the **Return/Enter** key.

- A password prompt is displayed.

7 Enter *sybase password*, then press the **Return/Enter** key.

- You are authenticated as yourself and returned to the UNIX prompt
- Your new home directory is **/usr/ecs/OPS/COTS/sybase** and all required environment variables have been set.

NOTE: GDAAC implementation has a sybsetup.csh file (type source sybsetup.csh).

11 Type, **isql –U<username>** and press the **Return/Enter** key.

12 Type the *password* when prompted for it and press the **Return/Enter** key.

13 Type **shutdown SYB_BACKUP** at the "**1>**" prompt and press the **Return/Enter** key.

14 Type **go** at the "**2>**" prompt and press the **Return/Enter** key.

4.1.6 Stop the ASE Server

1 Use **shutdown** to bring the server to a halt.

- This command can only be issued by the Sybase System Administrator (sa).
- If you do not give a server name, shutdown shuts down the ASE Server you are using.
- Syntax:

    1> shutdown [backup_server_name]] [with] [wait] [with nowait]
    2> go

2   To see the names of the backup servers that are accessible from your ASE Server, execute **sp_helpserver**.

- Use the value in the name column in the shutdown command.
- You can only shut down a Backup Server that is:
  - Listed in **sysservers** on your ASE Server
  - Listed in your local interfaces file

4.2 Creating Database Devices and Logical Volumes

Table 4.2-1, below, provides an Activity Checklist for creating database devices and logical volumes.

Table 4.2-1. Creating Database Devices and Logical Volumes - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Create a Database Device	(P) 4.2.1	
2	DBA	Convert Raw Partitions into XLV	(P) 4.2.2	
3	DBA	Perform Backout	(P) 4.2.3	

4.2.1 Create a Database Device

Database devices store the objects that make up databases. The term "device" can refer to either a physical or a logical device. Logical devices include any portion of a disk, such as a *partition* or a *file* in the file system used to store databases and their objects. A database device can be created when the System Administrator determines that new disk space is available for database use or as part of the recovery process. The System Administrator makes a request to the Data Base Administrator (DBA), who creates the new database device and notifies the System Administrator when the device has been created.

In order to create a new device, the DBA must have the following:

- The name of database device to be created
- A physical device on which to place database device
- The device size in megabytes
- The name of the mirror device, if one is in effect

*The creation of a database device is the mapping of physical space to a logical name and virtual device number (**vdevno**) contained in the master database.* The **disk init** command is used to initialize this space. **Disk init** syntax is:

name = "device_name"
physname = "physicalname"
vdevno = virtual_device_number (optional: will use the next available number)

> **size = number_of_blocks**
> **[vstart = virtual_address**
> **cntrltype = controller_number] (optional)**

Before you run **disk init**, see the server installation and configuration guide for your platform for information about choosing a database device and preparing it for use with the server. You may want to repartition the disks on your computer to provide maximum performance for your databases. **Disk init** divides the database devices into allocation units of 256 2K pages, a total of 1/2MB. In each 256-page allocation unit, the **disk init** command initializes the first page as the allocation page, which will contain information about the database (if any) that resides on the allocation unit.

Once initialization is complete, the space described by the physical address is available to the server for storage and a row is added to the **sysdevices** table in the master database. The initialized database can be:

- Allocated to the pool of space available to a user database
- Allocated to a user database and assigned to store a specific database object or objects
- Used to store a database's transaction logs

NOTE: Unless you are creating a small or non-critical database, always place the log on a separate database device.

After you run the **disk init** command, be sure to use **dump database** to dump the master database. This makes recovery easier and safer in case master is damaged. If you add a device and fail to back up master, you may be able to recover the changes with **disk reinit**.

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press the **Return/Enter** key.

3   Log into the server on which the new database device is to be created by typing: **/tools/bin/ssh** *<ServerName>*, then press the **Return/Enter** key.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears, type your *Passphrase* and then press the **Return/Enter** key.
- Go to step 6.

5   At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the **Return/Enter** key.

NOTE: For each database device to be created, perform Steps 6 through 9.

6 Using the text editor of your choice, edit *DeviceName*.**sql** and make changes to information enclosed in brackets (including the brackets) as appropriate:

- An example of an **add_devices.sql** file for creation of a database device is shown in Figure 4.2.1-1.

```
/****************************************************/
/* name: [add_devices.sql]                          */
/* purpose:                                         */
/* written:                                         */
/* revised:                                         */
/* reason:                                          */
/****************************************************/
disk init name = [device name],
physname = "/dev/[device name]",
vdevno = [#],
size = [size]
go
sp_helpdevice [device name]
go
```

Figure 4.2.1-1. Example of an add_devices.sql File for Creation of a Database Device

- In the area delimited by **/* */**, enter an appropriate description of the script including the file name, date written and person who wrote the script, its purpose, and any other information deemed appropriate. Be sure to enclose each line of the comment between **/* */**.
- The **disk init name** is the *DeviceName* is used in Step 8 above. You may not use spaces or punctuation except the underscore character (_) in *DeviceName*. Remember that the name you assign is case-sensitive. Be sure there is a comma after *DeviceName*
- The physname is the *FullPath_to_DeviceName*. Be sure to enclose *FullPath_to_DeviceName* in double quotes. Be sure to place a comma after *FullPath_to_DeviceName* but outside the double quotes.
- The **vdevno** is the *VirtualDeviceNumber*. **vdevno** is a unique identifying number for the database device. It must be unique among all the devices used by ASE Server and is never reused. Device number 0 represents the device named **d_master** that stores the system catalogs. Legal numbers are between 1 and 255, but the highest number must be one less than the number of database devices for which your system is configured. For example, for a system with

the default configuration of 10 devices, the legal device numbers are 1-9. The default of 10 devices can be changed using **sp_configure**. For the GDAAC implementation, the next vdevno is located in the /usr/dba0x/server.devices/next.vdevno file, it should be incremented when used. If this file does not exist, the next available number can be determined by looking at the output from the **sp_helpdevice** command and selecting the next number in sequence. If you use an existing number, Sybase will return the message, "device activation error."

- The **size** is the *DeviceSize* in blocks. To compute the number of blocks, multiply the device size in megabytes by 512; e.g., a 1,000 Mb device has 512,000 blocks

9 After the changes have been made, save the file according to the rules of your text editor.

10 At the UNIX prompt, type **isql –U***username* **-S***ServerName* **-i***DeviceName*.**sql -o***Device Name.out* then press the **Return/Enter** key.
- *ServerName* is the name of the server on which the database device will be created.
- *DeviceName.sql* is the name of the script file you created in step 8.
- *DeviceName.out* is the filename of the script's output for confirmation and/or troubleshooting purposes.)
- The system will prompt you for a password.

11 At the **Password:** prompt, type the *<password>* then press the **Return/Enter** key.
- When the UNIX prompt is again displayed the process is complete.

12 At the UNIX prompt, type **more** *DeviceName.out*, then press the **Return/Enter** key.
- This allows you to view the *DeviceName.out* file to confirm that the device has been created or to check for device creation errors.
- A sample of a completed database device creation script is shown in Figure 4.2.1-2.

```
/****************************************************/
/* name: test_dev.sql                               */
/* purpose: allocate 3Mb device for testing         */
/* written: 12/18/97                                */
/* revised:                                         */
/* reason:                                          */
/****************************************************/
disk init name = test_dev,
physname                                           =
     "/usr/ecs/Rel_A/COTS/sybase/studentdevices/test_dev.dat",
vdevno = 15,
size = 1536
go
sp_helpdevice test_dev
go
```

Figure 4.2.1-2. Completed Database Device Creation Script

4.2.2 Convert Raw Partitions into XLV

Logical volume disk driver (XLV) devices provide access to disk storage as logical volumes. *A logical volume is an object that behaves like a disk partition, but its storage can span several physical disk devices*. Using XLV, disks can be concatenated disks to create larger logical volumes, stripe data across disks to create logical volumes with greater throughput, and plex (or mirror) disks for reliability. In addition, XLV allows volume configurations changes to be made volumes are actively being used as a filesystem.

The geometry of logical volumes (e.g., the disks that belong to it, how they are put together, etc.) is stored in the disk labels of the disks that belong to the logical volumes. When the system starts up, the utility **xlv_assemble(1M)** scans all the disks on the system and automatically assembles them into logical volumes. **xlv_assemble(1M)** also creates any necessary device nodes.

Device numbers range from 0 to one less than the maximum number of logical volume devices configured in the system. This is 10 by default. This number can be changed by rebuilding a kernel with **lboot(1M).** There is a kernel driver, referred to as **xlv**, and some daemons for the logical volume devices. The driver is a pseudo device not directly associated with any physical hardware; its function is to map requests on logical volume devices into requests on the underlying disk devices. The daemons take care of error recovery and dynamic reconfiguration of volumes.

XLV allows the DBA to work with whole volumes and pieces of volumes. Pieces of volumes are useful for creating and reconfiguring volumes in units that are larger than individual disk partitions.

Each volume consists of up to three subvolumes. An **xfs(4)** filesystem usually has a large data subvolume in which all the user files and metadata such as inodes are stored and a small log subvolume in which the filesystem log is stored. For high-performance and real-time applications, a volume can also have a real-time subvolume that contains only user files aligned at configurable block boundaries. Guaranteed rate I/O can be done to real-time subvolumes.

1   Back up all databases.

2   BCP the **syslogins** table from master.

3   bcp master..syslogins out file.out -Usa -P -c

4   bcp master..sysloginroles out file.out -Usa -P -c

5   Save all information regarding the **sysdevices** and database options by executing the following command: **sp_helpdb db_name -- each database sp_helpdevice sp_helpdb**

6   Shutdown the backup and ASE servers.

7   After the partitions have been updated to XLV, **chown** the sybase disks to sybase:users. Also, change the **/etc/init.d/sybase** script.

8   Clean up some old Sybase files from **$SYBASE/devices** and **$SYBASE/ASE-12_5install**.

9   Execute **srvbuild** to initialize a NEW ASE server/backup (same server name).

10   Change sa password. (**sp_password NULL,newpass**)

11   Increase the number of devices. (**sp_configure "number of devices",20**)

12   Alter the database for master and tempdb.

13   Change **sysservers** to name the server as well as the new backup server. **sp_addserver nodename_srvr,local,nodename_srvr** , **sp_addserver SYB_BACKUP,local,nodename_backup**

14   Execute scripts to initialize the devices and to create databases (for load).

15   Load database dumps.

16 Verify that database data and log are not on same device. If on same device, update **sysusages** by deleting the log that uses the data device and then update the size of the data device to the full size (plus the one used by the log).

17 Uncompress the load file manually.

18 Online the database.

19 BCP in the **syslogins** table.

20 BCP in the **sysloginroles** table.

21 Change the DB dbo.

22 Reset all DB options.

23 Dump all databases.

4.2.3 Perform Backout

The backout procedure is executed only if it is necessary to restore the old ASE server because of a XLV conversion failure.

1 After restoring the sybase raw disk partitions from XLV partitions, change the ownership of the devices to **sybase:user**. This step has to be executed by someone with root access.

2 Login as sybase and cd to the **$SYBASE/ASE-12_5/install** directory.

3 Execute **srvbuild** and configure a new ASE server.
- Use the same configurations that were used from original ASE server, i.e., port numbers.

4 Once you have configured a new server, get into isql and execute the following:
- Change sa password.
- Alter database master to same size as before.
- Modify the sysservers with the ASE and backup names.
- Shutdown the ASE server and bring it up in single user mode (**-m** option).
- Uncompress the master database dump (do not load at this time).
- Change directory to **/usr/ecs/OPS/COTS/sybase/** and copy the old **sybsystemprocs.dat** to the new one.
- Load the dump of the master database.
  - After the load, it would automatically shut down the ASE server, so start it up as normal and not as single user.

5   If your databases do not recover at this point, mark them as suspect.

6   Check your devices and load each database from the dumps according to the following
          steps:
- Check all the devices and make sure that they exist.
- Drop the databases using **dbcc dbrepair** (**dbname,dropdb**) option.
- Re-create the databases with a **for load** option.
- Load each of the databases and do an online command after
- Verify that your **database** options were set properly as well as all the users on each
  database

7   Once everything is back to its original state, do a complete backup of all your databases.

4.3 Installing Databases and Patches

Table 4.3-1, below, provides an Activity Checklist for installing databases and patches.

Table 4.3-1. Installing Databases and Patches - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Perform a Database Build (Example)	(P) 4.3.1	
2	DBA	Install a Database Patch (Example)	(P) 4.3.2	

4.3.1 Perform a Database Build (Example)

The ECS Assistant is a custom application that simplifies the process of installing, testing and managing system software. The **Subsystem Manager** screen is used in the operational environment. The **Database** option is used to install, drop, patch, and update subsystem-specific databases.   The **Install** option is used to install custom software in a particular mode. The **Configuration** option is used to create CFG, ACFG and PCFG files for selected components. The **Stage Area Installation** option is used to input the staging location where the delivered software is stored. The **View Task Output** option is used to view results as the specified task is executing.   A detailed description of ECS Assistant use can be found in 609-EMD-001, *Release 7.11 Operations Tools Manual for the EMD Project*.

To perform a database build follow the steps included in the documentation (e.g., release notes) that specifies performing the build. The following steps are an example of a database that was built and populated with data from the version of the database that the new database was replacing.   Unless otherwise specified, the scripts that were run are located in the **/usr/ecs/<MODE>/CUSTOM/dbms/<SUBSYSTEM>** directory.

1   Create new db logins for new servers.
        • For example:
            **# EcDsStDb6ALogins <MODE> <dbo_ user_ name> <dbo_ user_ password>
            <server_ name> <DBNAME>**

2   Run the database verification script on the existing database.
        • For example:
            # EcStDbVerify5B <MODE> <USERNAME> <SERVER> <DBNAME>

        • The script is located in the **/usr/ecs/<MODE>/CUSTOM/dbms/COM/DBAdmin**
          directory. The output from this script is located in
          /usr/ecs/<MODE>/CUSTOM/logs and is named
          EcStDbVerify5B.<DBSERVER>.<DBNAME>.<DATETIME>.

3   Bulk copy data out of the existing database:
        • For example:
            **# EcDsStBulkCopyOut5B <MODE> <USERNAME> <PASSWORD>
            <SERVER> <DBNAME>**

        • The output from this script is saved as
          /usr/ecs/<MODE>/CUSTOM/logs/EcDsStBulkCopy.log.

4   Build the database using **ECS Assistant**.
        • Using the **SubSystem Manager**, execute the **DbBuild** function to build the database.
        • Enter the appropriate parameters in the **Database Script Parameters** dialogue box.

5   Execute the **EcDbDDMUpdateVersionTable.sql** script file that is under the directory
        /usr/ecs/{MODE}/CUSTOM/dbms/COM/DBAdmin.
        • The parameters to be passed are **MODE USERNAME DSQUERY DBNAME**.
        • You will be prompted for the password.

6   Review the file **EcMsDbBuild.log** in **/usr/ecs/<MODE>/CUSTOM/logs** for any error or
        warning messages.

7   Have the DBA verify the database changes through the **DbDesc** script for this subsystem.

8   Verify that the database version identifiers are correct.
        • For example:
            **# isql -S <server_ name> -U <db_ user_ name> -P <db_ user_ password>
            # use <database>_< MODE>
            # go
            # select * from EcDbDatabaseVersions where EcDbCurrentVersionFlag=" Y"
            # go**

9   Run a database transition upgrade script.

- For example:

  # EcDsStDbSybaseUpgrade. ksh \<MODE> \<USERNAME> \<PASSWORD> \<SERVER> \<DBNAME>

10   Run a script to create all of the new tables required for transition into the new database.

- For example:

  # EcDsStCreate5BTranTables \<USERNAME> \<PASSWORD> \<SERVER> \<DBNAME>

11   Bulk copy data in the old tables into the new database.

- For example:

  # EcDsStBulkCopyIn5B \<MODE> \<USERNAME> \<PASSWORD> \<SERVER> \<DBNAME>

- The output from this script is saved as
  /usr/ecs/\<MODE>/CUSTOM/logs/EcDsStBulkCopy.log.

12   Run the Transition Script.

- For example:

  # EcDsStTransition5Bto6A \<MODE> \<USERNAME> \<PASSWORD> \<SERVER> \<DBNAME>

  NOTE: You will need to have the secure shell passphrase and install login password ready to enter when prompted.

- The output from this script is saved as
  /usr/ecs/\<MODE>/CUSTOM/logs/EcDsStTransition.log.

13   Run the database verification script (script is located in the
/usr/ecs/\<MODE>/CUSTOM/dbms/COM/DBAdmin directory).

- For example:

  # EcStDbVerify6A \<MODE> \<USERNAME> \<SERVER> \<DBNAME>

- The output from this script is saved as
  /usr/ecs/\<MODE>/CUSTOM/logs/EcStDbVerify6A.\<DBSERVER>.\<DBNAME>.\<DATETIME>.

14   Check the differences between the database verification script output logs for the old database and the new database:

- For example:

  # diff -w EcStDbVerify5B.\<DBSERVER>.\< DBNAME>.\<DATETIME> EcStDbVerify6A.\<DBSERVER>.\<DBNAME>.\<DATETIME>

15   Compare the output of this statement with the differences expected based on information in the reference document.

16   Run a script to remove all tables required for transition into the new database:
- For example:

  # EcDsStDrop5BTranTables <USERNAME> <PASSWORD> <SERVER> <DBNAME>

4.3.2 Install a Database Patch (Example)

To install a database patch follow the steps included in the documentation (e.g., release notes) that specifies installing the patch. To install database patches, perform the following steps for all subsystems/components and then perform the appropriate subsystem/component-specific procedures.

1   Verify the current version of the database being patched.

 # isql -S <server_ name> -U <db_ user_ name> -P <db_ user_ password>
 # use <db_ name>[_< MODE>]
 # go
 # select * from EcDbDatabaseVersions where EcDbCurrentVersionFlag=" Y"
 # go

2   Compare the current database version against the appropriate version listed in the table provided in the patch instructions.
- If the current version is greater than or equal to the appropriate version listed in the table, continue with the database patch.
- If the current version is less that the appropriate version listed in the table, stop and patch the deficient database.

3   From the **ECS Assist Subsystem Manager** select the appropriate mode, subsystem, and component from the main window.

4   From the **ECS Assist Subsystem Manager** select **DbPatch** from the **Database** menu.
- A **File Selection** window appears.

5   From the **ECS Assist Subsystem Manager** (in the **File Selection** window) select **.dbparms** and **OK**.

6   Follow subsystem-specific installation instructions included in the documentation (e.g., release notes) that specifies installing the patch to complete the database patch process.
- For example:
  1   Logon to the host where the subsystem database package is installed.

2   Start ECS Assist's Subsystem Manager, select the appropriate mode, subsystem, and component.

3   Select **DbPatch** from the **Database** menu. A **File Selection** window appears.

4   In the "**File Selection**" window, select "**.dbparms**" and then "**Ok**". The "**Configurable Database Parameters**" dialog box appears.

5   Verify the patch number (referring to the table in the patch instructions). If it is not correct, correct it, enter the required information and select **Ok**.

4.3.3 COTS Databases

AutoSys and Remedy have Sybase databases. COTS installation is performed by the auto-install applications in the **/usr/ecs/OPS/COTS** directory. The results of the installation are stored in the appropriate subdirectories, e.g., an **autosys_install.scr** file located in the AutoSys home directory (**/use/ecs/OPS/COTS/autosys**).

4.4 Configuring Databases

Table 4.4-1, below, provides an Activity Checklist for configuring databases.

Table 4.4-1. Configuring Databases - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Configure the ASE Server	(P) 4.4.1	
2	DBA	Display Configuration Parameters	(P) 4.4.2	

4.4.1 Configure the ASE Server

Configuration parameters are user-defined settings that control various aspects of a database server's behavior. The server supplies default values for all configuration parameters. However, some of the configuration parameters need to be customized (refer to *SYBASE SQLServer 11.0.x: ALL DAAC Database Configurations,* 910-TDA-021).

Configuration parameters are grouped according to the area of server behavior that they affect. This makes it easier to identify all parameters that may need to be tuned in order to improve a particular area of ASE Server performance. The groups are:

- Backup/Recovery.
- Cache Manager.
- Disk I/O.
- General Information.
- Languages.
- Lock Manager.
- Memory Use.

- Network Communications.
- Operating System Resources.
- Physical Memory.
- Processors.
- ASE Server Administration.
- User Environment.

While each parameter has a primary group to which it belongs, many have secondary groups to which they also belong. For instance, the parameter number of remote connections belongs primarily in the Network Communications group, but also secondarily in the ASE Server Administration group and the Memory Use group. This reflects the fact that some parameters have implications for a number of areas of ASE Server behavior.

The configuration parameters are divided between two tables:

- Sybase Configuration Parameter Table.
- DAAC-Specific Configuration Parameter Table.

The Sybase Configuration Parameter Table groups the configuration parameters by functionality. It provides a list of configurable parameters under each functional group, and provides recommended values for these parameters:

The following type of information is captured in the Sybase Configuration Parameter Table:

- Group Name.
- Group Description.
- Parameter Name (for each group).
- Parameter Description.
- Command Line.
- Status (Static or Dynamic parameter).
- Recommended Value.
- Range (Minimum and maximum parameter value).
- Display Level (Basic, Intermediate, Comprehensive).
- Impact (Brief description of type of impact when a parameter value is configured).
- Controlling Authority (Entity responsible for controlling changes to a configuration parameter).
- Comments (This will include suggestions and examples).

Configuration parameters can be set or changed in one of two ways:

- By executing the system procedure **sp_configure** with the appropriate parameters and values.
- By hand-editing your configuration file and then invoking **sp_configure** with the configuration file option.

Configuration parameters are either dynamic or static. *Dynamic parameters go into effect as soon as* **sp_configure** *is executed. Static parameters require the ASE server to reallocate memory and take effect only after ASE Server has been restarted.*

The roles required for using **sp_configure** are as follows:

- Any user can execute **sp_configure** to display information about parameters and their current values.
- Only System Administrators and System Security Officers can execute **sp_configure** to modify configuration parameters.
- Only System Security Officers can execute **sp_configure** to modify values for allow updates, audit queue size and remote access.

The system procedure **sp_configure** displays and resets configuration parameters. One can restrict the number of parameters **sp_configure** displays by using **sp_displaylevel** to set the display level to one of the three values: Basic, Intermediate, or Comprehensive.

Figure 4.4.1-1 shows a sample **sp_configure** output. **Name** column is the description of the variable. The **minimum** column is the minimum allowable configuration setting**. Maximum** is the theoretical maximum value to which the configuration option can be set. The actual maximum value is dependent on the specific platform and available resources to the ASE server. **Config_value** reflects the current system default values. **Run_value** reflects the values the system is currently using, which may be different from the default values.

ASE Server configuration can be done either interactively by using **sp_configure** or non-interactively by invoking sp_configure with a configuration file. Configuration files can be used for several reasons:

- To replicate a specific configuration across multiple servers by using the same configuration file.
- To use a configuration file as a baseline for testing configuration values on your server.
- To use a configuration file to do validation checking on parameter values before actually setting the values.
- To create multiple configuration files and to switch between them as your resource needs change.

Previous releases of ASE Server required that reconfigure be executed after executing **sp_configure**. This is no longer required. The reconfigure command still exists, but it no longer has any effect.

```
name                             minimum      maximum    config value  run value
----------------------------  -----------  -----------  ------------  -----------
recovery interval                       1        32767             0            5
allow updates                           0            1             0            0
user connections                        5   2147483647             0           25
memory                               3850   2147483647             0         5120
open databases                          5   2147483647             0           12
locks                                5000   2147483647             0         5000
open objects                          100   2147483647             0          500
procedure cache                         1           99             0           20
fill factor                             0          100             0            0
time slice                             50         1000             0          100
database size                           2        10000             0            2
tape retention                          0          365             0            0
recovery flags                          0            1             0            0
nested triggers                         0            1             1            1
devices                                 4          256             0           10
remote access                           0            1             1            1
remote logins                           0   2147483647             0           20
remote sites                            0   2147483647             0           10
remote connections                      0   2147483647             0           20
pre-read packets                        0   2147483647             0            3
upgrade version                         0   2147483647          1002         1002
default sortorder id                    0          255            50           50
default language                        0   2147483647             0            0
language in cache                       3          100             3            3
max online engines                      1           32             1            1
min online engines                      1           32             1            1
engine adjust interval                  1           32             0            0
cpu flush                               1   2147483647           200          200
i/o flush                               1   2147483647          1000         1000
default character set id                0          255             1            1
stack size                          20480   2147483647             0        28672
password expiration interval            0        32767             0            0
audit queue size                        1        65535           100          100
additional netmem                       0   2147483647             0            0
default network packet size           512       524288             0          512
maximum network packet size           512       524288             0          512
extent i/o buffers                      0   2147483647             0            0
identity burning set factor             1      9999999          5000         5000
allow sendmsg                           0            1             0            0
sendmsg starting port number            0        65535             0            0
```

Figure 4.4.1-1. Example of sp_configure Output

Configuration files are not automatically backed up when you back up your master database. As they are part of the operating system files, they should be backed up in the same way other operating system files are backed up.

In order to perform the following procedure, the DBA must have obtained the following information:

- Name of server to be configured
- New values for configuration variables.

To set ASE Server configuration variables, the DBA must have **sa_role** (ASE Server) permissions. To set ASE Server configuration variables **allow updates**, **audit queue size**, **password expiration interval**, or **remote access**, **sso_role** (ASE Server) is also required.

Some parameter values take effect as soon as you reset the value. Others, which involve memory reallocation, do not change until you reset the value and then reboot ASE Server.

1   Log into the server that will be reconfigured by typing: **telnet** *ServerName* or **rlogin** *ServerName*, then press **Return**.

2   If a **Login**: prompt appears, log in as yourself by typing: *YourUserID*, then press **Return**.
     • A password prompt is displayed.

3   Enter *YourPassword* then press **Return**.
     • You are authenticated as yourself and returned to the UNIX prompt.

4   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press **Return**.

5   Begin the ASE session by typing at the UNIX prompt **isql -S***ServerName* then press **Return**.

6   Type **sp_configure** then press **Return**.

7   Type **go**, then press **Return.**
     • A list of the configurable parameters, their maximum and minimum values, the current setting and the default values is displayed.

8   After determining which parameter(s) to reset, type: **sp_configure *"ParameterName"***, *NewValue*, then press **Return**.
     • *ParameterName* is the name from the list displayed in Step 7 above.
        – Be sure to enclose the name in double quotes.
     • *NewValue* is the numeric value that you want to assign to *ParameterName*.

9   Type **go**, then press **Return.**

10   Repeat Steps 7 through 9 for each parameter you want to reconfigure.

11   When complete, type **quit** at the isql prompt, then press **Return**.
     • You are returned to the UNIX prompt.

4.4.2 Display Configuration Parameters

There are many configurable parameters associated with the custom software. Some of them are set by default to values that may be appropriate for most operating conditions. Others may be set to values that may or may not be appropriate for the requirements of operations at a particular

DAAC. Some parameters may be changed using system Graphical User Interfaces (GUIs) specifically designed to monitor and control functions related to particular subsystems. Others may require changes to a configuration file or database.

NOTE: Before changing any configuration parameter, make certain either that it is not under configuration control or that you have obtained any necessary approval specified in local Configuration Management policies and procedures.

Configuration parameters for custom software are managed by the Configuration Registry. The Configuration Registry Server provides a single interface to retrieve configuration attribute-value pairs for servers from the Configuration Registry Database, via a Sybase Server. The Configuration Registry Server maintains an internal representation of the tree in which configuration attribute-value pairs are stored. General configuration parameters used by many servers are stored in higher nodes in the tree. Parameters specific to a single server are contained in the leaf nodes of the tree. A script tool is available for loading the Configuration Registry database from data in configuration files. This loading is a one-time event to populate the Registry database with the information contained in **.cfg** files. Once the Configuration Registry is loaded, if the configuration files are moved or otherwise made inaccessible to the software, the software goes to the Configuration Registry to obtain needed configuration parameters. There is also a Configuration Registry application that can be used to view and edit configuration data in the database. Changes to the Configuration Registry typically are under the control of Configuration Management and the Database Administrator.

Data in the Registry database are stored hierarchically, sorted by hostnames. An application may have multiple entries, each under a different hostname where it runs. The application queries the database directly. The application shows the Attribute Tree on the left. It displays parameters and their values in the Attribute Listing on the right. It provides the operator with capability to create, read, update, and delete database entries. A click on a parameter name in the Attribute Listing displays a pop-up window, permitting the user to update or delete the parameter.

Use the following procedure to display configuration parameters.

1   On workstation *x0dms##*, in a terminal window, type
     /usr/ecs/*mode*/CUSTOM/utilities/EcCsRegistryGUIStart *mode* & at a UNIX
     command prompt and then press the **Return/Enter** key (where *mode* is likely to be **TS1**,
     TS2, or **OPS**).
     • The *x* in the workstation name is a letter designating your site: **g** = GSFC, **m** = SMC,
       l = LaRC, **e** = EDC, **n** = NSIDC, **o** = ORNL, **a** = ASF, **j** = JPL; the *##* is an
       identifying two-digit number (e.g., **g0dms03** indicates a data management subsystem
       workstation at GSFC). If you access the workstation through a secure shell remote
       login (ssh), you must enter **setenv DISPLAY <local_workstation IP address>:0.0**
       prior to the ssh before entering the command after the ssh. The *<ipaddress>* is the ip
       address of *x0mss##*, and **xterm** is required when entering this command on a Sun
       terminal.

- The Database Login window is displayed with entries filled in for **User Id:** (e.g., **EcCsRegistry**), **Server:** (e.g., *x0icg02_srvr*), and **DB Name:** (e.g., **EcCsRegistry_*mode***).

2   In the Database Login window click in the **Password:** field and type the password.
- The typed password is not displayed (dots are displayed in place of the password).

3   Click on the **Sign On** button.
- The Database Login window is closed and the Configuration Registry GUI is displayed.

4   On the tree showing system hosts displayed on the left side of the GUI, click on the "+" sign next to one of the hosts for which parameters are to be displayed.
- The tree displays a **config** branch.

5   Click on the "+" next to **config**.
- The tree displays a **CFG** branch.

6   Click on the "+" next to **CFG**.
- The tree displays the computer software components for the selected host.

7   Click on one of the listed components (or its folder icon).
- The **Attribute Listing** field displays the configuration parameters associated with the selected component.
- If there are a large number of parameters, the right side of the window will have a scroll bar that may be used to scroll down the list.

8   Click on one of the listed parameters.
- The **Attribute Information** pop-up window for the selected parameter is displayed, showing detailed information concerning the parameter.
- If you are logged in with an account authorized with appropriate permissions, the **Attribute Information** window permits changing or deleting the parameter.

9   To exit from the Configuration Registry GUI, follow menu path **File→Exit**.

4.5 Working with Indexes, Segments, and Caches

Table 4.5-1, below, provides an Activity Checklist for working with indexes, segments, and caches.

Table 4.5-1. Working with Indexes, Segments, and Caches - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Delete an Index	(P) 4.5.1	
2	DBA	Create Database Segments (Example)	(P) 4.5.2	

4.5.1 Delete an Index

An index provides a means of locating a row in a database table based on the value of a specific column(s), without having to scan all data in the table. When properly implemented, indexes can significantly decrease the time it takes to retrieve data, thereby increasing performance. Sybase allows the definition of two types of indexes:

- *Clustered index, where the rows in a database table are physically stored in sequence determined by the index.* The ASE Server continually sorts and re-sorts the rows of a table so that their physical order is always the same as their logical (or indexed) order. Clustered indexes are particularly useful when data are frequently retrieved in sequential order. Only one clustered index can be defined for a table.
- *Non-clustered indexes differ from their clustered counterpart in that the physical order of rows is not necessarily the same as their indexed order*. Each non-clustered index provides access to the data in its own sort order, giving the appearance of data in that physical order. Up to 249 non-clustered indexes can be defined for one table.

Clustered indexes allow faster searches than non-clustered indexes. Clustered indexes are often called the primary key of the table. If you don't specify which type of index you want, it will be a non-clustered index by default.

There are a lot of options for creating an index, but the most commonly used one is unique. Both clustered and non-clustered indexes can be unique. *A unique index prohibits duplicate values in the column that the index is on.* They cannot be created on text or image datatypes because those datatypes cannot reliably be declared unique.

Index keys need to be differentiated from logical keys. Logical keys are part of the database design, defining the relationships between tables: primary keys, foreign keys, and common keys. *When you optimize your queries by creating indexes, these logical keys may or may not be used as the physical keys for creating indexes.* You can create indexes on columns that are not logical keys, and you may have logical keys that are not used as index keys.

Tables that are read-only or read-mostly can be heavily indexed, as long as your database has enough space available. If there is little update activity, and high select activity, you should provide indexes for all of your frequent queries. Be sure to test the performance benefits of index covering.

Also, remember that with each insert, all non-clustered indexes have to be updated, so there is a performance price to pay. The leaf level has one entry per row, so you will have to change that row with every insert.

1   To delete an index, type **drop index**.

2   Type the table name that has the index, a period, and the index name that you want to delete.

3   Type **go** then press **Enter**.

4.5.2 Create Database Segments (Example)

Segments are named subsets of the database devices available to a particular ASE Server database. A segment can best be described as a label that points to one or more database devices.

Segmenting can increase ASE Server performance and give the SA or DBA increased control over placement, size and space usage of specific database objects. For example:

- If a table is placed on one device and its non-clustered indexes on a device on another disk controller, the time required to read or write to the disk can be reduced since disk head travel is usually reduced.
- If a large, heavily used table is split across devices on two separate disk controllers, read/write time may be improved.
- The ASE Server stores the data for text and image columns on a separate chain of data pages. By default, this text chain is placed on the same segment as the table. Since reading a text column requires a read operation for the text pointer in the base table and an additional read operation on the text page in the separate text chain, placing the text chain on a separate physical device can improve performance.

Segments are created within a particular database from the database devices already allocated to that database. Each ASE Server database can contain up to 32 segments. The database devices must first be initialized with **disk init** and then be made available to the database with a **create database** or **alter database** statement segment names can be assigned.

When a database is first created, the ASE Server creates three segments in the database:

- System, which stores the database's system tables.
- Logsegment, which stores this database's transaction log.
- Default, which initially stores all other database objects.

Additional segments can be created. Subsystem databases, for example, consist of the following:

- Default data segment used if no other segment specified in the create statement.
- SYSLOGS, transaction logs.
- System tables and indexes.
- OPS mode data segment.
- OPS mode index segment.
- TS1 mode data segment.
- TS1 mode index segment.

- TS2 mode data segment.
- TS2 mode index segment.

If tables and indexes are placed on specific segments, those database objects cannot grow beyond the space available to the devices represented by those segments, and other objects cannot contend for space with them. Segments can be extended to include additional devices as needed. Thresholds can be used to provide warnings when space becomes low on a particular database segment.

In the following example procedure, the DBA receives a request to create a segment for the storage of the SubServer table indexes in the t1ins01_srvr database on a separate physical disk. Two devices, **subserver_data** and **subserver_index**, have already been created and are located on different physical disks.

In order to perform the procedure, the DBA must have obtained the following information:
- Name of database
- Database device extents
- Space on the database allocated to the Database device

To create a database, the DBA must have **sa_role**.

1   At the UNIX prompt, type **cd ASE-12_5/scripts** and then press **Return**.

2   Using the text editor of your choice, edit *segment*.**sql** and make changes to information enclosed in brackets (including the brackets) as appropriate.

3   After the changes have been made, save the file according to the rules of the text editor.

4   At the UNIX prompt, type:**isql –U<username> -S**<*ServerName*> **-i**<*Segment*>**.sql -o**<*Segment*>**.out** and then press **Return.**
- The system will prompt you for a password.

5   At the **Password:** prompt, type the *<password>*, then press **Return.**
- When the UNIX prompt is again displayed the process is complete.

6   At the UNIX prompt, type **more** *Segment.out* , then press **Return**.
- This allows you to view the *Segment.out* file to confirm that the device has been created or to check for database creation errors.

Figure 4.5.2-1 shows an example of creation of a database segment template file.

```
/*******************************************/
/* name: [segment.sql]                     */
/* purpose:                                */
/* written:                                */
/* revised:                                */
/* reason:                                 */
/*******************************************/


 sp_addsegment [seg_name], [DBname],[Device Name]
gogo[DBDevice]
```

Figure 4.5.2-1. Example of Creation of a Database Segment Template File

After the segment has been defined in the current database, the **create table** or **create index** commands use the optional clause **"on segment_name"** to place the object on a particular segment. Syntax:

> **create table table_name (column_name datatype ...) [on segment_name]**
> **create [clustered | nonclustered] index index_name on table_name (columns)**

Use **sp_helpdb** database_name to display the segments defined for that database.

Use **sp_helpsegment** segment_name to list the objects on the segment and show the mapped devices.

4.5.3 Caches

A cache is a block of memory that is used by Sybase to retain and manage pages that are currently being processed. By default, each database contains three caches:

- Data caches retain most recently accessed data and index pages
- Procedure caches retain most recently accessed stored procedure pages
- User transaction log caches are transaction log pages that have not yet been written to disk for each user

The size of each of these default caches is a configurable item that must be managed on a DAAC-by-DAAC basis. These caches can be increased or decreased as needed. The data cache can be further subdivided into named caches. *A named cache is a block of memory that is named and used by the database management system (DBMS) to store data pages. The named data caches can be used only by databases or database objects that are explicitly bound to them. All*

objects not explicitly bound to named data caches use the default data cache. A database, table, index, or text or image page chain can be bound to a named data cache.

Assigning a database table to named cache causes accessed pages to be loaded into memory and retained. Named caches greatly increase performance by eliminating the time associated for disk input and output (I/O).

4.6 Backing Up and Recovering Data

Database and transaction log backups and recoveries are probably the most important tasks the DBA must perform. Without regular and complete backups of databases and transaction logs, the potential for enormous amounts of data to be lost is very great. How often you back up a database depends on how frequently the data is updated and how costly it would be to lose it. Databases that are constantly being modified need to be backed up frequently; databases containing relatively static data can be backed up less frequently. You should regularly back up production databases, for example. If you experience a media failure, you can restore the data from the most recent backups the **restore** command. You do not have to shut down servers to back up or restore a database. The Backup Server makes it possible to perform online backup and restore operations while the Adaptive Server is running.

Table 4.6-1, below, provides an Activity Checklist for backing up and recovering data.

Table 4.6-1. Backing Up and Recovering Data - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Perform Automatic Backups	(P) 4.6.1	
2	DBA	Perform Manual Backups	(P) 4.6.2	
3	DBA	Perform a User Database Recovery (Order of Procedures)	(P) 4.6.3	
4	DBA	Perform a User Database Recovery (Example)	(P) 4.6.4	
5	DBA	Create a User Database (Example)	(P) 4.6.5	

4.6.1 Perform Automatic Backups

Each DAAC is responsible for determining its own backup schedule to meet its individual requirements. It is strongly suggested that a regular schedule of database backups be established and maintained. Although the databases are included in the daily backups of the entire system, separate database backups should be performed nightly.

Database backups are run by a UNIX **cron** job which executes the **E.CoSyb_DumbDb** program located in the **/usr/ecs/CUSTOM/dbms/COM/DBAdmin** directory. No intervention in the automatic backup process is required by the DBA, though periodic checks of the backup subdirectories are recommended.

Manual backups can be performed at any time by the DBA and are recommended for the following situations:

- Any change to the master database, including new logins, devices, and databases.
- Any major change to user databases, such as a large ingest or deletion of data, definition of indexes, etc.
- Other mission-critical activities as defined by the DAAC operations controller.

The following are procedures and scripts files that are currently being used for Sybase backups. There are **cron** jobs running at all Sybase servers that have ASE server installed. All dump files are currently written to local machine. The site DBA is responsible for configuring the backup dump to the remote Sybase directory.

1   Check if **crontab** is up and running.

       > crontab -l

- The output will look something like the following:

  019 * * 1-6 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_DumpDb
  012 * * 1-6 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_DumpTran
  021 * * 1-5 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_CkErrorLog

2   If **crontab** is not running, enter:

       > crontab /usr/ecs/OPS/COTS/sybase/run_sybcron

- The files shown in Table 4.6-2 will be installed by ECS Assistant to the directory **/usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin**:

Table 4.6-2. Automatic Backup Files (1 of 2)

Script	Description
EcCoDbSyb_SetupKsh	This file contains the SYBASE and DSQUERY (server) environment setup. This file is called by EcCoDbSyb_DumpDb, EcCoDbSyb_DrumpTran, and EcCoDbSyb_CkErrorLog scripts.
EcCoDbSyb_DumpDb	This script contains the code to dump the databases. First, it checks for any DBCC error on the master database, if there is any error on the master, the script sends an email to the DBA and exits the program. If the master database dump was successful, then the rest of the databases are dumped. Each database has a DBCC check; if there is any error on the database then the database is NOT dumped and an email is sent to the DBA. At the end, a status email is sent, providing all the database names that were successfully dumped.

Table 4.6-2. Automatic Backup Files (2 of 2)

Script	Description
EcCoDbSyb_DumpTran	This script contains the code to dump the transaction logs. This dumps the transaction logs for each database, it checks the error log file; if the error Msg is 4207 or 4221 it does a dump of the database first, then it does the transaction dump. If there is any other error Msg then the transaction dump fails and email is sent. At the end, the status of the transaction log dumps is emailed to the DBA.
EcCoDbSyb_SedFile	This file contains all the database that don't need to be dump (i.e., temp, model, etc.)
EcCoDbSyb_DboMail	This file contains the email list of all the DBAs.
EcCoDbSyb_DbStat	This script updates the index table of a database. This script is called from EcCoDbSyb_DumpDb after each successfully database dump.
EcCoDbSyb_CkErrorLog	This script checks for specific database error messages from the Sybase Error Log File every hour and emails the error messages to the DBAs in the EcCoDbSyb_DboMailfile.
EcCoDbSyb_tran_log.awk	This script matches the current hour with the hour the error messages were generated in the Error Log File. If errors found, the messages are saved in a mail file and sent to DBAs.
0 0 * * 1-6 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_DumpDb	
0 10,13,16 * * 1-6 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_DumpTran	
0 * ** 1-6 /usr/ecs/OPS/CUSTOM/dbms/COM/DBAdmin/EcCoDbSyb_CkErrorLog	
NOTE: Make sure there is an OPS mode directory with all script files.	

3   Modify the files shown in Table 4.6-3 before running any of the scripts shown in Table 4.6-2.

Table 4.6-3. Files That Need to Be Modified before Running Scripts

File	Modification
EcCoDbSyb_SetupKsh	Make user you have the SYBASE files under /usr/ecs/OPS/COTS/sybase
EcCoDbSyb_SedFile	Add any other database that might not need to be backed up. The databases that are listed in this file do not need to be backed up.
EcCoDbSyb_DboMail	Add/delete the email of the DBA and any other email that might need to be added/deleted. All the errors and status are sent to them.

ASE Server backups are performed nightly by a **cron** job which runs the **run_sybcron** program located in the **$SYBASE**/ directory. The following table of definitions (Table 4.6-4) is used throughout the rest of this section.

Table 4.6-4. Automatic Backup Components

Name	Function
run_sybcron	File added with the crontab -e command, contains several executable cron commands. **Example:** 00 19 * * 1-6 /data1/COTS/sybase/sybasedump
EcCoDbSyb_DumpDb	Controlling script that performs the following functions: run isql to create the Backup Statements run isql to execute the Backup Statements record the results of the Backup Statements in Backup Files copy the Backup Files to the Backup Subdirectory create the Backup Summary greps successful Dump statements along with any errors generated, sends e-mail to the DBA and writes them to the backup_summary file
sp	ASE Server password file - contains password for backup role

4.6.2 Perform Manual Backups

Both the **dump database** and **dump transaction** command processing are off-loaded to the backup server and will not affect normal operations of the database. These commands are performed by the System Administrator on appropriate databases using the following syntax.

> 1> **dump database master to
> "/usr/ecs/OPS/COTS/sybase/sybase_dumps/dumps/dbname.dat.MMDDHHMM."**

> 2> **go**

After dumping the database, compress the dump file by executing:

> **%compress /usr/ecs/OPS/COTS/sybase/sybase_dumps/dumps/dbname.dat.MMDDHHMM**

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY *IPNumber*:0.0** or **setenv DISPLAY *ServerName*:0.0**, then press **Return**.

3   Log into the server that contains the database to be backed up by typing: **/tools/bin/ssh** *gsfcsparc5.gsfcmo.ecs.nasa.gov*, then press **Return**.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '*<user@localhost>*'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '*<user@localhost>*'** appears, type your *Passphrase* and then press the **Enter** key.
- Go to step 6.

5   At the *<user@remotehost>***'s password:** prompt, type your ***Password*** and then press the **Enter** key.

6   Log into Sybase by typing: **su-sybase**, then press **Return.**
   - A password prompt is displayed.

7   Enter ***SybasePassword*** then press **Return**.
   - Remember that ***SybasePassword*** is case-sensitive.
   - You are authenticated as yourself and returned to the UNIX prompt.
   - Your new home directory is **/usr/ecs/OPS/COTS/sybase** and all required environment variables have been set.

8   At the UNIX prompt, type **isql -Usa** then press **Return**.

9   To backup the database, at the isql prompt, type **dump database *DBName* to *BackupDirectory/DBName*.dat**, then press **Return**.

10   Type **go**, then press **Return**.

11   To backup the transaction log, at the isql prompt, type **dump transaction *DBName* to *BackupDirectory/DBName*_tx.dat**, then press **Return**.

12   Type **go**, then press **Return**.

4.6.3 Perform a User Database Recovery (Order of Procedures)

Database recovery is performed when the Database Administrator determines that some database data is corrupt or when the System Administrator determines that a database device has failed. The System Administrator makes a request to the Database Administrator, who performs the restore and notifies the System Administrator when the restore is complete.

The symptoms of media failure are as variable as the causes. If only a single block on the disk is bad, your database may appear to function perfectly for some time after the corruption appears; this is why you should run **dbcc** commands frequently. If an entire disk or disk controller is bad, you will not be able to use a database. ASE Server marks it as suspect and displays a warning message. If the disk storing the master database fails, users will not be able to log into the server, and users already logged in will not be able to perform any actions that access the system tables in master.

The complete database device restoration procedure is actually a suite of procedures that must be performed in the prescribed order:

1   The device failure has been verified by the System Administrator and restoration has been requested from the DBA.

2 If possible, perform a dump of the current database and the current database transaction log for each database on the failed device to be restored. If this is not possible due to the damage to the database device, then the DBA will have to use the most recent database and transaction log dumps.

3 If possible, the DBA examines the space usage for each database on the failed device. The same defaults should be set when the new database device(s) is (are) created.

4 The DBA drops the database(s) on the failed device, then the database device itself is dropped.

5 The DBA initializes a new database device. This is why it is important to keep the device creation scripts.

6 The DBA recreates each user database on the new database device. This is why it is important to keep the user creation scripts.

7 Each database is reloaded with data from the database backups and the transaction log backups. It is this procedure that is detailed below.

8 The DBA notifies the System Administrator when the database restoration is complete.

In order to perform the procedure, the DBA must have obtained the following information:

- Name of database device that has failed
- Name of the replacement device
- Name of the databases that reside on the failed device
- Name of the backup volumes
- Name of the dump files on the backup volumes

Manual recovery of a user database is performed by the System Administrator by the use of the **load database** and **load transaction** commands. For issues concerning the **master** database, please consult your System Administrator's Guide for assistance. It is recommended that any user database to be recovered be dropped and created with the **for load** option.   The **databasename**.sql along with any **alter.databasename**.sql scripts can be combined into one script which will re-create the user database with the **for load** option. This will insure the success of the **load database** and **load transaction** commands.

4.6.4 Perform a User Database Recovery (Example)

In the example procedures that follow, the database **UserDB** was created using the following script excerpt (stored in home/scripts/create.databases/userdb.sql):

create database UserDB on data_dev1 = 100 log on tx_log1 = 50 [with override]

and was modified using the following script excerpt (home/scripts/create.databases/alteruserdb.sql):

> **alter database UserDB on data_dev1=50**

For the purposes of this example, the full database backup and transaction log dumps were successful and located in **/usr/ecs/OPS/COTS/UserDB.dat** and **UserDB_tx.dat**.

1   In the $SYBASE/ASE-12_5/scripts/ directory, make a script file to load database and save template for future use.
- Syntax:

> **% cd /usr/ecs/OPS/COTS/sybase/scripts/,**
> **% cp template.sql userdb_for_load.sql.**

NOTE: If template does not exist, vi userdb_for_load.sql.

2   Modify appropriate items so that the script file resembles the following:

> **1> create database UserDB on data_dev2=100 log on tx_log2=50 for load**
> **2> go**
> **3> alter database UserDB on data_dev3=50**
> **4> go**

3   Save the script in **$SYBASE**/ASE-12_5**/scripts/create.databases/userdb_for_load.sql**

4   Run the script from the UNIX command prompt.

> **%isql -Usa -Sservername -iuserdb_for_load.sql -ouserdb_for_load.out**

5   Check the **userdb_for_load.out** file for success.

6   Load the database from the full backup.

> **1> load database UserDB from**
> **"/usr/ecs/OPS/COTS/sybase/sybase_dumps/week1/dbname.dat.MMDDHHMM"**
> **1> go**

7   Load the transaction file from the transaction file dump.

> **1> load transaction UserDB**
> **from"/usr/ecs/OPS/COTS/sybase/sybase_dumps/week1/dbname.tran.MMDDHHMM"**
> **2> go**

4.6.5 Create a User Database (Example)

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press **Return**.

3   Log into the server on which the new database device is to be created by typing: **/tools/bin/ssh** *<ServerName>*, then press **Return**.
   - If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears; continue with Step 4.
   - If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears, type your *Passphrase* and then press the **Enter** key. Go to step 6.

5   At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the **Enter** key.

NOTE: For each database device to be created, perform Steps 6 through 9:

6   At the UNIX prompt, type **cd** /ASE-12_5/installthen press **Return.**
   - This places you in the directory that contains all the scripts used to create database devices.
   - It is important that these scripts not be deleted from the system in case it is necessary to restore a database device following a failure (see **Restore Databases**).

7   Using the text editor of your choice, edit *DBName***.sql** and make changes to information enclosed in brackets (including the brackets) as appropriate (see Figure 4.6.4-1).
   - **DBName** is the name of the database and should describe its function succinctly.
   - **DBDeviceName** is the name of the existing, approved database device on which **DBName** will reside.
   - **DBSize_in_MB** is the estimated size of the database in megabytes.
   - **LogDBDeviceName** is the name of the database device holding the transaction log
   - **LogSize_in_MB** is the estimated size of the transaction log in megabytes.
   - The **sp_helpdb** command provides feedback at the end of the script to assure that the database has been created.

```
/* *********************************************** */
/* name: template.sql                     */
/* purpose:                          */
/* written:                      */
/* revised:                        */
/* reason:                          */
/* *********************************************** */
create database database_name
on data_dev = size, /* in MB */
log on log_dev = size
go
sp_helpdb database_name

go
```

Figure 4.6.4-1. Sample template.sql File for Creation of a Database

8   After the changes have been made, save the file according to the rules of the text editor.

9   At the UNIX prompt, type:

isql –U<username> -S<ServerName> -i*DBName*.sql   -o*DBName*.out

then press **Return**.

- *ServerName* is the name of the server on which the database device will be created.
- *DBName.sql* is the name of the script file you created in step 8.
- *DBName.out* is the filename of the script's output for confirmation and/or troubleshooting purposes. The system will prompt you for a password.

10   At the **Password:** prompt, type the *<password>* then press **Return.**
- When the UNIX prompt is again displayed the process is complete.

11   At the UNIX prompt, type **more *DBName.out*** then press **Return**.
- This allows you to view the *DBName.out* file to confirm that the device has been created or to check for database creation errors.

A sample of a completed **Create Database** script is shown in Figure 4.6.4-2.

```
/*****************************************/
/* name: test_db.sql                     */
/* purpose: create a database for testing */
/* written: 12/19/97                     */
/* revised:                              */
/* reason:                               */
/*****************************************/
create database test_db
on test_dev = 3
go
sp_helpdb test_db
go
```

Figure 4.6.4-2. Completed Create Database Script

4.7 Establishing Database Security

Permissions control access within a database. There are two types of permissions within a database: object and command. Object privileges control select, insert, update, delete, and execute permissions on tables, views, and stored procedures. Command permissions control access to the create statements for databases, defaults, procedures, rules, tables, and views. Access controls allow giving various kinds of permissions to users, groups, and roles with the **grant** command. The **revoke** command rescinds these permissions.

The ability to assign permissions for the commands is determined by each user's status (e.g., system administrator, database owner, database object owner) and by whether or not a particular user has been granted a permission with the option to grant that permission to other users.

Groups are a means of logically associating users with similar data access needs. Once a group has been defined, object and command permissions can be granted to that group. A user who is member of a group inherits all of the permissions granted to that group.

Roles provide a structured means for granting users the permissions needed to perform standard database-related activities and also provide a means for easily identifying such users. They provide a means of enforcing and maintaining individual accountability. There are six pre-defined roles shown in Table 4.7-1 that may be assigned to a user. Other unique roles can be created and granted to users, groups of users, or other roles. In addition, role hierarchies can be defined. This enables a user assigned one role to automatically have the privileges available to all other roles lower in the hierarchy. Mutually exclusive roles can also be defined. This can be done in terms of membership – a single user cannot be granted both roles – or activation – a single user can be granted two roles but cannot enable both (e.g., both cannot be enabled simultaneously).

Table 4.7-1. Roles and Privileges

Roles		Privileges
System Administrator	sa_role	Grant a specific user permissions needed to perform standard system administrator duties including: Installing SQL server and specific ASE server modules Managing the allocation of physical storage Tuning configuration parameters Creating databases
Site Security Officer	sso_role	Grant a specific user the permissions needed to maintain ASE server security including: • Adding server logins • Administrating passwords • Managing the audit system • Granting users all roles except the sa_role
Operator	oper_role	Grant a specific user the permissions needed to perform standard functions for the database including: • Dumping transactions and databases • Loading transactions and databases
Navigator	navigator_role	Grant a specific user the permissions needed to manage the navigation server
Replication	replication_role	Grant a specific user the permissions needed to manage the replication server
Sybase Technical Support	sybase_ts_role	Grant a specific user the permissions needed to execute database consistency checker (dbcc), a Sybase supplied utility supporting commands that are normally outside of the realm of routine system administrator activities

Table 4.7-2, below, provides an Activity Checklist for establishing database security.

Table 4.7-2. Establishing Database Security - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Grant or Revoke Database Access Privileges	(P) 4.7.1	
2	DBA	Create an ASE Server Login	(P) 4.7.2	
3	DBA	Change a Password	(P) 4.7.3	
4	DBA	Perform Auditing	(P) 4.7.4	

4.7.1 Grant or Revoke Database Access Privileges

The DBA uses the **grant** and **revoke** statements to control permissions. The syntax for grant and revoke statements is similar:

> **grant {all [privileges] | command_list} to {public | name_list | role_name}**
> **revoke {all [privileges] | command_list} from { public | name_list | role_name}**.

1   Determine the privileges that you want to grant and to which user(s).

2   Log on to a local host.

3   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press **Return**.

4   Log into the server where the user database resides by typing: **/tools/bin/ssh** *ServerName*, then press **Return**.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears; continue with Step 5.
- If you have not previously set up a secure shell passphrase, go to Step 6.

5   If a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears, type your *Passphrase* and then press the **Enter** key.
- Go to step 7.

6   At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the **Enter** key.

7   At the UNIX prompt, type **su-sybase** then press **Return**.
- A password prompt is displayed.

8   Enter *SybasePassword*, then press **Return**.
- You are authenticated as yourself and returned to the UNIX prompt.
- Your new home directory is **/usr/ecs/OPS/COTS/sybase** and all required environment variables have been set.

9   At the UNIX prompt, type **isql -Usa**, then press **Return**.

10   At the **Password:** prompt, type the *SybaseSAPassword* then press **Return.**
- Note: the *SybaseSAPassword* is case-sensitive.

11   If you are granting privileges, at the isql prompt, type **grant** *Privilege* **to** *LoginName* then press **Return**.
- If more than one privilege is to be granted, repeat this step on subsequent lines.

12   If you are revoking privileges, at the isql prompt, type **revoke** *Privilege* **from** *Logname*
           then press **Return**.
    - If more than one privilege is to be revoked, repeat this step on subsequent lines.

13   When all privileges have been assigned, type **go** then press **Return**.

NOTE: It is recommended that the DBA store the privilege granting and revoking
                commands in **.sql** files in the **$SYBASE**/ASE-12_5/**scripts/..** directory along with
                their results.

4.7.2 Create an ASE Server Login

A user is given a login account with a unique ID. All of that user's activity on a server can be attributed to his or her server user ID and audited.   Passwords are stored in the **master..syslogins** table in encrypted form. When users log in from a client, they can choose client-side password encryption before sending the password over the network. A user can be granted the ability to impersonate another user. This proxy authorization allows administrators to check permissions for a particular user or to perform maintenance on a user's database objects. Application servers can execute procedures and commands on behalf of several users.

Providing users with access to ASE servers and their databases consists of the following steps:

    - A server login account for a new user is created.
    - The user is added to a database and optionally assigned to a group.
    - The user or group is granted permissions on specific commands and database objects.

The user needs an ASE Server login to access the DBMS. The login is assigned to a user with the related permissions. During initial database installation, logins used by the custom code were created and permissions assigned for access to subsystem databases. In addition, special database installation logins, **<SUBSYS>_role**, were created to support database installation needs. For each login, the level of access is limited to that associated with their login, group or assigned group/role. Object permissions are set within the installation scripts of the <SUBSYS> subsystem for each object and group/role.

In order for a user to connect to an ASE Server, a login must be created by the DBA. That login must then be assigned to particular database(s) that the user will access. All login details are stored in the syslogins table in the **master** database. Two stored procedures are required:

    - The stored procedure **sp_addlogin** adds new login names to the server but does not grant access to any user database. Syntax:
      sp_addlogin login_name, password, [,default database ,language, fullname]
    - The stored procedure **sp_adduser** adds the user. Syntax:
      1> **sp_adduser**login_name [username, group_name]
      2> **go**

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press **Return**.

3   Log into the server for which a new login procedure is to be created up by typing: **/tools/bin/ssh** *ServerName*, then press **Return**.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '***<user@localhost>***'** appears, type your *Passphrase* and then press the **Enter** key. Go to step 6.

5   At the *<user@remotehost>***'s password:** prompt, type your *Password* and then press the **Enter** key.

6   At the UNIX prompt, type **su-sybase** then press **Return**. A password prompt is displayed.

7   Enter *SybasePassword* then press **Return**.
- Note: *SybasePassword* is case-sensitive.
- You are authenticated as yourself and returned to the UNIX prompt.
- Your new home directory is /usr/ecs/OPS/COTS/sybase and all required environment variables have been set.

8   At the UNIX prompt, type **cd** ASE-12_5**scripts/server.users** then press **Return**.

9   Using the text editor of your choice, edit *UserName***.sql** (see Figure 4.7.2-1) and make changes to information enclosed in brackets (including the brackets) as appropriate:

10   After the changes have been made, save the file according to the rules of the text editor.

```
/****************************************/
/* name: [template.sql]                 */
/* purpose:                             */
/* written:                             */
/* revised:                             */
/* reason:                              */
/****************************************/
     sp_addlogin [login name], [password], [default database]
     go
     use [default database]
     go
     sp_adduser [login name]
     go
     /* the following is optional, by database */
     sp_changegroup [group name], [login name]
     go
     sp_helpuser
go
```

Figure 4.7.2-1. Sample template.sql File for New Database User Login

11   At the UNIX prompt, type **isql -Usa -S***ServerName* **-i***UserName***.sql -o***UserName***.out**
           then press **Return**.
           • *ServerName* is the name of the server on which the master database is stored.
           • *UserName***.sql** is the name of the script file you created in Step 9.
           • *UserName***.out** is the filename of the script's output for confirmation and/or
             troubleshooting purposes.
           • The system will prompt you for a password.

12   At the Password: prompt, type the *SybaseSAPassword* then press **Return**.
           • When the UNIX prompt is again displayed the process is complete.

13   At the UNIX prompt, type **more** *UserName***.out** then press **Return**.
           • This allows you to view the *UserName***.out** file to confirm that the user login has
             been created or to check for user login creation errors.

4.7.3 Change a Password

One of the most common problems a DBA encounters is a user who cannot connect to an ASE
Server. The following procedure is applicable.

1   At the UNIX prompt, type **isql -U***dbastudent* then press **Return.**

- The system will prompt you for a password.

2   At the **Password:** prompt, type the *dbastudentpassword* then press **Return.**

3   At the Sybase prompt, type the following:

sp_password *old-password, new-password*

NOTE: The dba (or any user with sso_role) may change another user's password with the following syntax: (this is the most common activity performed)

sp_password *sso_role password, new-password, login name*

4.7.4 Perform Auditing

A comprehensive audit system is provided with Adaptive Server. The audit system consists of a system database called sybsecurity, configuration parameters for managing auditing, a system procedure, **sp_audit**, to set all auditing options, and a system procedure, sp_addauditrecord, to add user-defined records to the audit trail. When you install auditing, you can specify the number of audit tables that Adaptive Server will use for the audit trail. If you use two or more audit tables to store the audit trail, you can set up a smoothly running audit system with no manual intervention and no loss of records.

A System Security Officer manages the audit system and is the only user who can start and stop auditing, set up auditing options, and process the audit data. As a System Security Officer, you can establish auditing for events such as:

- Server-wide, security-relevant events.
- Creating, deleting, and modifying database objects.
- All actions by a particular user or all actions by users with a particular role active.
- Granting or revoking database access.
- Importing or exporting data.
- Logins and logouts.

The following statements represent examples for performing auditing:

1   Run installsecurity auditing located under $SYBASE/ASE-12_5/scripts.

2   Add a login for auditing:

> **sp_addlogin ssa, ssa_password, sybsecurity**
> **use sybsecurity**
> **sp_changedbowner ssa**
> **sp_role "grant", sso_role, ssa**

3   Enable auditing:

   sp_auditoption "enable auditing", "on"
   sp_auditlogin loginname, "cmdtext", "on"

4   Create a table in a database with one field.

5   Grant all on the table for the **loginname.**

6   Log into isql using the **loginname.**

7   Insert a record into the table.

8   Log into isql as **ssa.**

9   Select * from sysaudits where **loginname** = **"loginname"**

NOTE: Once auditing is turned on, the **sysaudit** tables will get filled up very quickly. The database option options of **trunc log on chkpt** needs to be turned on in the **sybsecurity** database so that the **logsegment** of this database can be cleaned up upon checkpoint. This **logsegment** can also be cleaned up by means of transaction dumps if the database is installed on two separate disk devices. Also, as soon as auditing is turned on, a **cron** job needs to be turned on at the same time to **bcp** out and truncate the **sysaudit** table. This is very important.

4.7.5 EMD Security Directive

All System Administrators and Database Administrators at the sites are responsible for reasonable security measures when installing custom software. This means:

- Changing the permissions of online secure files to the minimum level required.
- Backing up secure file(s) to removable media (floppy or tape) and removal of secure files immediately after installation is complete and then keeping the removable medium in a secure location.

The following file is affected as result of this requirement on the EMD program:

   /usr/ecs/<MODE>/CUSTOM/dbms/<SUBSYSTEM>/Ec<server>SybaseLogins.sql

Set permissions to 711 (user read, write, execute; group and other read only).

4.8 Copying Individual Databases

To create an exact copy of a database:

- Dump the existing database.
- Create a database to load with this dump.

The new database does not have to be the same size as the original.   The only requirement is that the destination database must be at least as large as the dumped database and have the same

beginning fragments as the original database. This information can be obtained from saved database creation scripts or by running the following command:

select segmap,'Size in MB'=size/512 from sysusages where dbid= db_id("database_name")

4.8.1 Copying a Database (Example)

A database was created with the following statement:

create database dbname on datadevice1 = 1000,
log on Logdevice1 = 200
go
alter device dbname on datadevice2 = 500   running:
select segmap,'Size in MB'=size/512 from sysusages
where dbid= db_id("dbname")
would return:segmap Size in MB
3   1000
4   200
3   500

You could create a 3GB database as follows and load your database into it (using **for load** option will shorten database load time):

create database newdatabase on datadevice3 = 1000 log on logdevice3 = 200
for load
go
alter database newdatabase on datadevice 3=500 for load go
alter database newdatabase on datadevice4=300 for load go
alter database newdatabase on datadevice5=1000 for load go
load database newdatabase from dbname_dump go

4.9 Bulk Copying

Table 4.9-1, below, provides an Activity Checklist for bulk copying.

Table 4.9-1. Bulk Copying - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Perform Bulk Copying	(P) 4.9.1	

4.9.1 Perform Bulk Copying

The bulk copy (**bcp)** utility is located in the **$SYBASE/OCS-12_5/bin** directory and is designed to copy data to and from ASE Server databases to operating system files. You must supply the following information for transferring data to and from ASE Server:

- Name of the database and table.

- Name of the operating system file.
- Direction of the transfer (in or out).

In order to use **bcp**, you must have an ASE Server account and the appropriate permissions on the database tables and operating system files that you will use. To copy data **into** a table, you must have **insert** permission on that table. To copy data **out** to an operating system file, you must have **select** permission on the following tables:

- The table being copied:
    - **sysobjects**
    - **syscolumns**
    - **sysindexes**

bcp syntax:

> **bcp [[database_name].owner.]table_name {in | out} datafile [-e errfile] [-n] [-c]**
> **[-t field_terminator] [-r row_terminator] [-U username] [-S server]**

1   Log on to a local host.

2   Set display to current terminal by typing: **setenv DISPLAY** *IPNumber***:0.0** or **setenv DISPLAY** *ServerName***:0.0**, then press **Return**.

3   Log into the server that contains the database to be backed up by typing: **/tools/bin/ssh** *ServerName*, then press **Return**.
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears; continue with Step 4.
- If you have not previously set up a secure shell passphrase, go to Step 5.

4   If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, type your *Passphrase* and then press the **Enter** key.
- Go to step 6.

5   At the <*user@remotehost*>**'s password:** prompt type your *Password* and then press the **Enter** key.

6   Set Login as sybase and cd to /usr/ecs/OPS/COTS/sybase/scripts/.
- Note: *SybasePassword* is case-sensitive.
- You are authenticated as yourself and returned to the UNIX prompt.
- Your new home directory is **/usr/ecs/OPS/COTS/sybase** and all required environment variables have been set.

8   Modify the [table name]_out script with the correct database name, table name, direction, and login name and save.

9   Run the [table name]_out script and press **Return** for each of the prompts except the last.

10   At the last prompt store [your responses] in a [table name].fmt file. This creates a format file for future bulkcopy activity.

11   To copy the data to another already created table, repeat Steps 6, 7, and 8 with the following changes:
- The direction is "in"
- Use the optional **-f [format file name]**

4.10 Replication System Administration

Administering the replication system is primary the role of the Replication System Administrator (RSA). The Replication System Administrator installs, configures, and administers the replication system. Given the distributed nature of the MSS implementation this role may be performed by different people at different locations. If this is the case, various tasks for administering Replication Server may require coordination between Replication System Administrators.

The Replication System Administrator has **sa** user permissions, which provides that person with the ability to execute nearly all commands in the replication system. In managing the system, the Replication System Administrator may need to coordinate with DBAs for both local and remote databases.

Replication System Administrators should be experienced Sybase DBAs and should have taken the Sybase training classes *Replication System Administration* and *Replication Disaster Recovery Workshop*. They should also have read and understood the manuals: *Replication Server Administration Guide*, *Replication Server Configuration Guide for UNIX Platforms*, *Replication Server Reference Manual*, and *Replication Trouble Shooting Guide*.

4.10.1 Replication System Administrator Tasks

Table 4.10-1 identifies tasks required to maintain a replication system:

Table 4.10-1. Replication System Administrator Tasks

Task	Roles
Installing Replication Server	Replication System Administrator (RSA)
Adding or removing a Replication Server	RSA
Starting up and shutting down Replication Server.	RSA
Configuring Replication Server	RSA
Maintaining Routes (Creating and modifying)	RSA
Managing the RSSD	RSA
Adding a primary and replicate database.	RSA
Adding login names, database users, and administering appropriate permissions	RSA
Adding replicated tables or changing table schemas. Creating and modifying replicated tables Creating and modifying replication definitions Creating and materializing subscriptions at replicate sites.	RSA
Defining data server function-string classes and function strings.	RSA
Applying database recovery procedures.	RSA
Maintaining and monitoring database connections	RSA
Monitoring Replication Server	RSA
Processing rejected transactions	RSA
Quiescing Replication Server	RSA
Reconciling database inconsistencies.	RSA

4.10.2 DAAC DBA Replication Roles and Tasks

The Database Administrator (DBA) supports some Replication Server administrator tasks. Table 4.10-2 lists tasks that the DBA administrators perform at the local DAACs with respect to replication server administration.

Table 4.10-2. DAAC DBA Replication Roles and Tasks (1 of 2)

Task	Roles
Installing Replication Server	DBA
Managing the RSSD	DBA
Adding a primary and replicate database.	DBA
Adding login names, database users, and administering appropriate permissions	DBA
Adding replicated tables or changing table schemas. Creating and modifying replicated tables Creating and modifying replication definitions Creating and materializing subscriptions at replicate sites.	DBA

Table 4.10-2. DAAC DBA Replication Roles and Tasks (2 of 2)

Task	Roles
Defining data server function-string classes and function strings.	DBA
Applying database recovery procedures.	DBA
Processing rejected transactions	DBA
Quiescing Replication Server	DBA
Reconciling database inconsistencies.	DBA

4.10.3 Database Replication

The MSS database is the only database to use replication. This is done so user profile and order tracking data can be shared among the DAACs.

Replication begins with definitions and subscriptions. *Replication definitions specify the tables and columns that may be replicated for a data recipient. Replications subscriptions specify which definitions the data recipient wishes to receive.* In addition, the replication database and server must be configured to support the potentially large volumes of data that will be transferred between the source and recipient databases.

The MSS replication definition and subscription scripts were developed as templates for installation at DAACs and the SMC. Since peer-to-peer configuration for MsAcUsrProfile is required, the template approach was decided so that only those scripts that need to be implemented at each site are configured. The template files provide the necessary generic functions needed to configure the MSS MsAcUsrProfile replication environment. Unix shell scripts have been developed to allow installers to pass the appropriate site-specific information (i.e. database name, replicate replication server name, etc.) when prompted. The Unix script then customizes the template script files for the specific site. The DD&M group has created naming conventions for all replication scripts and naming conventions for the subscriptions, replication definitions, and other objects related to the replication environment. Script names consist of the following conventions:

<action>.<SUBSYS>.<Replication Object>.<Primary Site ID>.sql.<Replicate Site ID>

4.11 Performance Monitoring, Tuning, and Problem Reporting

Databases are ubiquitous in the system, and performance has been addressed from the beginning of the project. Each database has an expected performance profile, including initial size, growth rates, transaction rates, and response times. Analysis of database performance generally takes a multi-tiered approach. At the lowest level, the data access patterns are analyzed and the database physical design is tailored for these access patterns. These activities occur during the design stage of the development. During the implementation phase, actual transactions are inspected to make sure that they are efficiently coded, and are consistent with the expected use of the database. Once implemented, the performance of the database is measured under various loads; performance is tuned by modifying various configuration parameters provided by the database

management system, and in some cases by modifying the database design or the data access code.

Actual performance can be assessed from a response time perspective or a capacity perspective. Response times can be measured in a variety of ways. For some of the system databases, a direct user interface is available that will provide a basis for assessing response time performance—for example, searches provided through the client. For these cases, the analytical task becomes the separation of the database component of the response time from all other contributors. Other system databases do not have a direct user interface because they serve system internal functions; an example is the Subscription database. Response time performance for these databases may be measured using test drivers r by capturing timing marks from instrumented code as test threads are run through the system. Database capacity—usually measured as the number of transactions that the database can handle per second—is usually tested with test drivers, although it can also be measured when the system is put through thread tests, operational tests, or stress tests. Sybase monitoring tools can be used to measure the throughput of the system during such tests, although they add their own overhead and reduce the maximum capacity achievable. The Sybase monitoring tool collects an enormous wealth of information about the use of Sybase resources—everything from hardware (CPUs, memory, disk channels) to Sybase internally configured resources (cache blocks, spin locks, indexes). These resources can then be tuned according to the usage patterns observed (including the possible addition of hardware).

4.11.1 Monitoring

The easiest way to monitor performance is through the Monitor Server. The Monitor Server provides specific performance data on cache usage, network traffic, device I/O, and locking activity. *Performance is the measure of efficiency of an application or multiple applications running in the same environment*. Performance is usually measured in one of two ways:

- *Response time, which is the time that a single task takes to complete.*
- Throughput, which is the volume of work completed in a fixed time period, e.g. transactions per second (tps).

Most of your tuning efforts should address the amount of time it takes for the server to respond to queries.

You can run **sp_sysmon** both before and after tuning Adaptive Server configuration parameters to gather data for comparison. You can also use **sp_sysmon** when the system exhibits the behavior you want to investigate. In many tests, it is best to start the applications, and then start **sp_sysmon** when the caches have had a chance to reach a steady state. If you are trying to measure capacity, be sure that the amount of work you give the server keeps it busy for the duration of the test. Many of the statistics, especially those that measure data per second, can look extremely low if the server is idle during part of the sample interval. In general, **sp_sysmon** produces valuable information when you use it:

- Before and after cache or pool configuration changes.
- Before and after certain sp_configure changes.
- Before and after the addition of new queries to your application mix.

- Before and after an increase or decrease in the number of Adaptive Server engines.
- When adding new disk devices and assigning objects to them.
- During peak periods, to look for contention.
- During stress tests to evaluate an Adaptive Server configuration for a maximum expected application load.
- When performance seems slow or the system behaves abnormally.

4.11.2 Tuning

Response time can be shortened by reducing contention and waits times, particularly disk I/O wait times; using faster components; and reducing the amount of time the resources are needed. In some cases, Adaptive Server is optimized to reduce initial response time, that is, the time it takes to return the first row to the user. This is especially useful in applications where a user may retrieve several rows with a query and then browse through them slowly with a front-end tool. Other ways of increasing performance are summarized by system level in Table 4.11-1.

Table 4.11-1. Tuning Options

Layers	Tuning Options
Application	Remote or replicated processing to move decision support off machine
	Stored procedures to reduce compilation time and network usage
	Minimum locking level that meets application needs
Database	Transaction log thresholds to automate dumps and avoid running out of space
	Thresholds for space monitoring in data segments
	Partitions to speed loading of data
	Devices to avoid disk contention, take advantage of I/O parallelism
Server	Tuning memory, most critical configuration parameters and other parameters
	Configuring cache sized and I/O sizes
	Scheduling batch jobs and reporting for off hours
	Reconfiguring parameters for shifting workload patterns
Devices	More medium-sized devices and more controllers for better I/O throughput
	Distributing databases, tables, and indexes for even I/O load across devices
	Segments, partitions for I/O performance on large tables used for parallel queries
Network	Configuring packet sizes to match application needs
	Configuring subnets
	Isolating heavy network uses
	Configuring for multiple network engines
Hardware	Configuring the housekeeper task to improve CPU use
	Configuring multiple data caches
Operating System	Choosing between riles and raw partitions
	Increasing memory size

4.12 Ensuring Database Quality

4.12.1 Integrity Monitoring

The Database Consistency Checker (**dbcc**) is a set of utility commands for checking the logical and physical consistency of a database. Use the **dbcc** commands:

- As part of regular database maintenance (periodic checks run by the System Administrator). These checks can detect, and often correct, errors before they affect a user's ability to use ASE Server.
- To determine the extent of possible damage after a system error has occurred.
- Before backing up a database.
- Because you suspect that a database is damaged. For example, if using a particular table generates the message "Table corrupt", you can use dbcc to determine if other tables in the database are also damaged.

The integrity of the internal structures of a database depends upon the System Administrator or Database Owner running database consistency checks on a regular basis. Two major functions of dbcc are:

- Checking allocation structures (the commands checkalloc, tablealloc, and indexalloc).
- Checking page linkage and data pointers at both the page level and row level (checktable and checkdb). The next section explains page and object allocation and page linkage.

The **dbcc** command is used in the database backup scripts to determine if the database is properly configured and without errors. If errors occur, the backup will not proceed and a message is sent to the DBA with notification of the problem.

4.13 Sybase Troubleshooting

Table 4.13-1, below, provides an Activity Checklist for Sybase troubleshooting.

Table 4.13-1. Sybase Troubleshooting - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Troubleshoot Chronic Deadlock	(P) 4.13.1	

4.13.1 Troubleshoot Chronic Deadlock

A deadlock (also known as a "deadly embrace") is a situation where two database processes are simultaneously attempting to lock data that the other holds. For example, two users (A and B) are updating the same table of data at the same time. User A holds a lock on Page 1 and requests a lock on Page 2. Meanwhile, user B holds a lock on Page 2 and has requested a lock on Page 1. Without intervention, these two jobs would never finish.

Sybase detects these situations, analyzes the two processes and automatically kills the process with less accumulated processor time. Sybase prints out an error message to the killed process and requests that the user re-submit the job. A Sybase error message similar to the following one is displayed:

> **Msg 1205, Level 13, State 1:**
> **Server 'SERVER', Procedure 'sp_whatever', Line 123:**
> **Your server command (family id #0, process id #99) was deadlocked with another process and has been chosen as deadlock victim. Re-run your command.**
> **(return status = -3)**

1   Turn on deadlock printing information and **sp_configure** "print deadlock information".

2   Recreate the problem.

3   As deadlocks occur, **tail -f the $SYBASE/ASE-12_5/install/errorlog** (or wherever the errorlog prints out to).
  - Detailed information is printed out (e.g., line numbers of code causing deadlocks, tables within Sybase).

4   Run dbcc traceon(3604) and dbcc page(#) on the page numbers printed by traceflag 1204.
  - dbcc page will report the tablename and the index that is being used.
  - It is helpful to see if it is a non-clustered or a clustered index.

5   Monitor locks and blocked processes during deadlock re-creation.
  - Use **sp_lock**, **sp_block** and **sp_blocker** to print out detailed information about locking and blocking.

6   Run **sp_sysmon** while re-creating the problem, and analyze the "Lock Management" section.
  - Advanced deadlock diagnosis can be conducted using:
    - **traceflags 602, 603**, which prints out information for deadlock prevention.
    - **traceflag 1205**, which puts a stack trace on the deadlock.
    - **traceflag 8203**, which displays statement and transaction locks.
    - **sp_configure** "deadlock checking period", "deadlock retries", which are configuration parameters related to deadlocking.
  - Common deadlock problems and solutions include:
    - **Contention on heap tables**. "Last page contention" problems occur where high insert rates always seem to go on the last page of a table. This creates what is known as a "heap table," which is bound to have performance problems. The end of the table (in terms of a page chain) is a hotspot of activity. If this is the cause of the deadlocks, either partition the table (which creates several insert points for data and eliminates the "heapness") or re-create your clustered index on a field that spreads inserts along the entire page chain of the table (i.e.,

clustered indexes on surrogate keys as opposed to last name or something similarly random). This can be seen in **sp_sysmon** output "Last page locks on heaps."

– **Poorly written key generation schemes**. Using a "select max(col)+1" method or high rate OLTP applications generating keys from a badly configured **key_storage** table system are always deadlock candidates. Never use the "select max+1" method. If you must use a **key_storage** solution (if you can't deal with identity gaps for example), ensure that the key_storage table is configured for Row Level Locking (in Sybase 11.9.2 and above) or has max_rows_per_page=1 (in Sybase 11.5 and 11.0).

– **Lock contention; sp_configure "number of locks"**. If you have configured too few locks for the system, severe lock contention can occur and deadlocking can be common. There is an entire section **of sp_sysmon** devoted to Lock management. Large updating within multi-user applications and long running transactions will definitely lead to deadlocking. Don't issue multiple update statements within the same transaction. If you must update multiple objects within the same transaction, make sure you consistently access the objects in the same order throughout the application. Updating or deleting from tables without covering indexes cause table scans. Other options: System data Lock strategies: In 11.9.2 and above: examine whether it may be wise to go to Row Level locking or Data only locking.

– **Hardware problems**. Slower I/O devices increase deadlock vulnerability.

– **Cursors**. Should be avoided.

– **Parallelism**. y using some of the parallel processing features available in later versions, deadlocking is subdued.

– **ANSI isolation level 3** (serialization) locking. Avoid when possible.

– **Descending scans in indexes**. Can sometimes cause deadlocks after upgrading where none previously occurred.

– **Application-side deadlocking**. See Section 1.5.1 of the Sybase FAQ for a quick example of Application-side deadlocking and how to avoid it.

– **Deadlocking increases with upgrade**. These deadlocks are frequently caused by the "allow backward scans" option being turned on by default. They can also be caused by differing optimizer paths with new version (after upgrade) or just the engine's sheer speed. Also recommended to increase lock escalation threshold if using RLL (defaults to 200, which is not a lot of rows).

4.13.2 Space Usage

Thresholds are defined on segments to provide a free space value at which a procedure is executed to provide a warning or to take remedial action. Use **sp_addthreshold** to define your own thresholds**: sp_addthreshold database_name, segment_name, free_space, procedure_name**, where **free_space** is the number of free pages at which the threshold

procedure executes; **procedure_name** is the stored procedure which the threshold manager executes when the number of free pages falls below the **free_space** value.

4.14 Oracle Procedures

Table 4.14-1, below, provides an Activity Checklist for performing Oracle procedures.

Table 4.14-1. Performing Oracle Procedures - Activity Checklist

Order	Role	Task	Section	Complete?
1	DBA	Start the Oracle Database	(P) 4.14.1	
2	DBA	Shut Down the Oracle Database	(P) 4.14.2	
3	DBA	Terminate a Session	(P) 4.14.3	
4	DBA	Control the Listener	(P) 4.14.4	
5	DBA	Specify Views	(P) 4.14.5	
6	DBA	Obtain Information and Control the System	(P) 4.14.6	
7	DBA	Perform Oracle Troubleshooting	(P) 4.14.7	
8	DBA	Access Dynamic Performance View	(P) 4.14.8	
9	DBA	Display Parameter Values	(P) 4.14.9	
10	DBA	Display Information about Users	(P) 4.14.10	
11	DBA	Display Information about System Privileges and Object Privileges	(P) 4.14.11	

The following items identify the location of oracle components:

- ORACLE_HOME - Specifies the directory where the Oracle software will be installed; e.g., /usr/ecs/OPS/COTS/oracle/8.1.6
- ORACLE_SID - Specifies the instance name.   SID - pds, pdsit, pdsst.
- ORACLE_BASE - /usr/ecs/OPS/COTS/oracle

4.14.1 Start the Oracle Database

To start the Oracle database, use the following procedure.

1   Log in to the appropriate host.

2   To start up the instance and open the database, type **STARTUP** and then press **Return**.

3   To start up the instance only, type **STARTUP NOMOUNT** and then press **Return.**

4   To change the state of the database from NOMOUNT to MOUNT, type **ALTER DATABASE MOUNT** then press **Return**.

5   To open the database, type **ALTER DATABASE OPEN;** then press **Return**.

4.14.2 Shut Down the Oracle Database

To shutdown the database and close the instance, the following procedure is applicable.

1   On the appropriate host, type **SHUTDOWN IMMEDIATE** and then press **Return**.

2   Type **SHUTDOWN** and then press **Return**.
- This is the normal shutdown. Oracle waits for all users to disconnect before completing the shutdown.

4.14.3 Terminate a Session

The following procedure is used to kill a user process or abnormal session.

1   **SELECT SID, SERIAL# FROM V$SESSION WHERE USERNAME=”*xxxxxxx*”;**

2   Write down the sid and serial# from above query.

3   Run **ALTER SYSTEM KILL SESSION ‘*integer1*, *integer2* ;**
- *integer1*: value of the SID column.
- *integer2*: value of the SERIAL#.
- When the ACTIVE user session is terminated, the transaction is rolled back and the user immediately receives the following message.

    ORA – 00028 : your session has been killed.

4.14.4 Control the Listener

The listener is the process on the server that listens to the connection request from the client's machine, e.g., pdssa. The following commands control the listener:
- Lsnrctl start = start up a listener.
- Lsnrctl stop =   shut down a listener.

4.14.5 Specify Views

The following commands specify views:
- DBA_xxx : Database Administrator's view.
- ALL_xxx : Expanded user's view.
- USER_xxx : User's view.

Example: Run a select statement for the Database Administrator's view of the following database information.

- General overview - DICTIONARY
- Schema objects view - DBA_OBJECTS, DBA_TABLES, DBA_TAB_COLUMNS, DBA_CONSTRAINTS
- Space allocation view - DBA_SEGMENT, DBA_FREE_SPACE, DBA_ENTENTS
- Database Structure - DBA_DATA_FILES, DBA_ROLLBACK_SEGS, DBA_TABLESPACES

4.14.6 Obtain Information and Control the System

There are many commands for managing the Oracle database. Table 4.14-2 lists a few that may be particularly useful.

Table 4.14-2. Some Useful Oracle Commands

Command	Function
ARCHIVE LOG LIST;	Displays the current ARCHIVELOG status of the database from within the Server Manager (the system must be in ARCHIVELOG mode to enable online or "hot" backups)
SELECT * FROM V$DATABASE;	SQL SELECT displays information from specified database table
SELECT GROUPS,CURRENT_GROUP#,SEQUENCE# FROM V$THREAD;	Displays information about log file groups
SELECT GROUP#MEMBERS,STATUS FROM V$LOG;	Display information about log file groups and members
ALTER SYSTEM SWITCH LOGFILE;	Forces Oracle to begin writing to a new redo log file group, regardless of whether the files in the current group are full
ALTER SYSTEM CHECKPOINT;	Forces Oracle to perform a checkpoint (i.e., ensures that all changes made by committed transactions are written to the data files on disk

4.14.7 Perform Oracle Troubleshooting

The following control file parameters need to be examined:

- BACKGROUND_DUMP_DEST - location where background process trace files are written. This is also the location for the alert log.
- DB_BLOCK_BUFFERS - Number of blocks cached in the SGA.
- SHARED_POOL_SIZE - Size in bytes of the shared pool.
- USER_DUMP_DEST - Location where user debugging trace files are created o behalf of a user process.

- PROCESS - Maximum number of operating system processes that can connect simultaneously to this instance.

Check each database's error log at the appropriate location:

- /usr/ecs/OPS/COTS/oracle/admin/pds/
- /usr/ecs/OPS/COTS/oracle/admin/pdsst/
- /usr/ecs/OPS/COTS/oracle/admin/pdsit/

4.14.8 Access Dynamic Performance View

Dynamic performance view can be accessed in the NOMOUNT stage. Run a select statement to view the dynamic database server performance.

- Select * from V$PARAMETER
- Select * from V$SGA
- Select * from V$PROCESS
- Select * from V$SESSION
- Select * from V$INSTANCE
- Select * from V$CONTROLFILE
- Select * from V$DATABASE
- Select * from V$DATAFILE
- Select * from V$LOGFILE
- Select * from V$TABLESPACE
- Select * from V$LOG

4.14.9 Display Parameter Values

To display a parameter value, type **SHOW PARAMETER;** then press **Return**.

4.14.10   Display Information about Users

To display information about users, type **SELECT USERNAME, ACCOUNT_STATUS FROM DBA_USERS;** then press **Return**.

4.14.11   Display Information about System Privileges and Object Privileges

The following commands (followed by a press of the **Return** key) are useful to obtain information about system privileges and object privileges:

- SELECT * FROM DBA_SYS_PRIVS;
- SELECT * FROM SESSION_PRIVS;
- SELECT * FROM DBA_TAB_PRIVS;
- SELECT * FROM DBA_COL_PRIVS;

5. Security Services

EMD security architecture must meet the requirements for data integrity, availability, and confidentiality. EMD Security Services meets these requirements by incorporating a variety of mechanisms to establish and verify user accounts, issue and verify passwords, audit user activity, and verify and protect data transfer. Security logs are monitored and security reports generated by the System Administrator as required. Several open source products provide tools for authentication and network and systems monitoring - Crack, ANLpasswd, TCP Wrappers, and Tripwire. Crack and ANLpasswd provide brute force password cracking and password checking, respectively for local system and network access. Tripwire monitors for intruders by noting changes to files. F-Secure Secure Shell (ssh) provides strong authentication access and session encryption from external, non-trusted networks as well as internally within a DAAC. Security Services also supports detection of, reporting, and recovery from security breaches. Security scans of each system are performed periodically to prepare for the formal security scans done biannually by the ESDIS IV&V contractor. These preliminary scans are done using the FOUNDSTONE Security Scanner product.

The following section defines step-by-step procedures for Operations personnel to run the Security Services tools. These procedures assume that DAAC Management has already approved the requester's application for a Security process. It is recommended that access to these tools be controlled through the **root access only**.

5.1 Scanning Network Vulnerabilities

The EMD project no longer has responsibility for scanning the network and network-attached systems. However, the FOUNDSTONE Security Scanner is a licensed product that NASA uses extensively to detect system level vulnerabilities. GSFC has a site license to use the product and any of the supported DAACs may use that license since all DAACs are using GSFC IP address space. This product does NOT belong to EMD and as such there is not an official release of it. A license key is required which can be obtained from the ESDIS Computer Security Official. The information the ESDIS Computer Security Official needs includes the IP addresses of the Production and M&O LANs. The software runs on Microsoft Windows NT or Microsoft Windows 2000. A laptop is the only practical way to run it. Once you have the key, the product is downloadable from the following URL:

https://secure.nai.com/apps/downloads/free-evaluations

You must register in order to get an account and password to download the product. The setup is like most PC products – run the setup.exe that you downloaded. It will query you for the license key.

ISS uses profiles, which tailor what and how it queries systems. To get the most recent version, contact the IV&V contractor, Titan Inc. at (301) 982-5414 and ask for the ESDIS security group.

5.2 Ensuring Password Integrity

One aspect of system security is discretionary access control based on user passwords. Passwords ideally would be so unique that they are virtually impenetrable to unauthorized users. Two products provide utilities to create effective password practices. "Crack" detects weak passwords that could be easily bypassed. It works in "batch" mode. ANLpasswd enforces strong password rules as the user is changing their password.

Crack and ANLpasswd provide a comprehensive dictionary, which can be shared. These "source" dictionaries provide lists of words, which if used, would create vulnerable passwords. You can add other dictionaries, for example, acronym lists, to eliminate commonly used terms from being used as passwords.

Crack is installed in a secure location that has **root access only**. ANLpasswd is automounted in /tools/bin.

5.2.1 Detecting Weak Passwords

Running Crack against a system's password file enables a system administrator to assess how vulnerable the file is to unauthorized users and how well authorized users select secure passwords. Crack is designed to find standard UNIX eight-character DES-encrypted passwords by standard guessing techniques.

Crack takes as its input a series of password files and source dictionaries. It merges the dictionaries, turns the password files into a sorted list, and generates lists of possible passwords from the merged dictionary or from information gleaned about users from the password file. It does not attempt to remedy the problem of allowing users to have guessable passwords, and it should NOT be used in place of getting a really good, secure password program replacement.

The instructions provided in the following sections are general in nature, because how you configure Crack is DAAC specific. Operations personnel should be familiar with these tasks to:

- Configure the Crack shell script and config.h files based on the README file and on requirements established for your site. See the Section on "Configuring Crack" below.
- Run Crack based on requirements established for your site. See "Running Crack" below.
- Customize the dictionaries. See Section "Creating Dictionaries" below.

5.2.1.1 Configuring Crack

Although Crack should already be configured for your system, the instructions are provided should you have to reconstruct the makefile as a result of file corruption. Crack has two configuration files: the Crack shell script, which contains all the installation-specific configuration data, and the file Sources/conf.h, which contains configuration options specific to various binary platforms. Use the following procedure for configuring crack.

1   In the Crack shell script, edit the CRACK_HOME variable to the correct value. This variable should be set to an absolute path name on which Crack will be run. (Path names relative to username are acceptable as long as you are using csh.)

- There is a similar variable, CRACK_OUT, which specifies where Crack should put its output files — by default, this is the same as $CRACK_HOME.

2   Edit the file Sources/conf.h and establish which switches to enable. Each #define has a small note explaining its purpose. Portability of certain library functions should not be a problem.

3   If using Crack-network (see Section 5.2.1.4, Options, below), generate a Scripts/network.conf file. This file contains:

- A list of hostnames that are rsh/ssh destinations.
- Their binary type (useful when running a network Crack on several different architectures).
- An estimate of their relative power (take your slowest machine as unary, and measure all others relative to it).
- A list of per-host flags to add to those specified on the Crack command line, when calling that host.
- There is an example of such a file provided in the Scripts directory.

4   To specify a more precise figure as to the relative power of your machines, play with the command *make* tests in the source code directory. This can provide you with the number of fcrypt()s that your machine can do per second. This number can be plugged into your *network.conf* as a measure of your machines' power (after rounding the value to an integer).

5.2.1.2 Running Crack

Crack is a self-installing program. Once the necessary configuration options for the Crack shell script and config.h have been set, the executables are created via *make* by running the Crack shell script.

NOTE: To run Crack on a YP password file, the simplest way is to generate a passwd format file by running the following command:

 # ypcat passwd > passwd.yp ↵

and then running Crack on the passwd.yp file.

To launch Crack:

1   To change directory, type **cd /usr/local/solaris /crack**, and then press the **Return/Enter** key.

2   To execute the program, type **./Crack**, and then press the **Return/Enter** key.

3   For the single platform version, type **./Crack [options] [bindir] /etc/passwd [...other passwd files]** and then press the **Return/Enter** key.

4   To execute over the network, type **./Crack -network [options] /etc/passwd [...other passwd files]** and then press the **Return/Enter** key.

For a brief overview of the [options] available, see Section 5.2.1.4, Options, below. Section 5.2.1.5, Crack Support Scripts, briefly describes several very useful scripts.

5.2.1.3 Creating Dictionaries

Crack works by performing several individual passes over the password entries that are supplied. Each pass generates password guesses based upon a sequence of rules, supplied to the program by the user. The rules are specified in a simplistic language in the files gecos.rules and dicts.rules, located in the Scripts directory (see Section 5.2.1.5, Crack Support Scripts, below).

Rules in Scripts/gecos.rules are applied to data generated by Crack from the pw_gecos and pw_gecos entries of the user's password entry. The entire set of rules in gecos.rules is applied to each of these words, which creates many more permutations and combinations, all of which are tested. After a pass has been made over the data based on gecos information, Crack makes further passes over the password data using successive rules from the Scripts/dicts.rules by loading the whole of Dicts/bigdict file into memory, with the rule being applied to each word from that file. This generates a resident dictionary, which is sorted and made unique to prevent wasting time on repetition. After each pass is completed, the memory used by the resident dictionary is freed up, and re-used when the next dictionary is loaded.

Crack creates the Dicts/bigdict dictionary by merging, sorting, and making unique the source dictionaries, which are to be found in the directory DictSrc and which may also be named in the Crack shell script, via the $STDDICT variable.   (The default value of $STDDICT is /usr/dict/words.)

The file DictSrc/bad_pws.dat is a dictionary that is meant to provide many of those common but non-dictionary passwords, such as 12345678 or qwerty.

To create your own dictionary:

1   Copy your dictionary into the DictSrc directory (use compress on it if you wish to save space; Crack will unpack it while generating the big dictionary).

2   Delete the contents of the Dicts directory by running Scripts/spotless. Your new dictionary will be merged in on the next run.

5.2.1.4 Options

Options available with the Crack command are:

-f Runs Crack in foreground mode, i.e., the password cracker is not put into the background, and messages appear on stdout and stderr as you would expect. This option is only really useful for very small password files, or when you want to put a wrapper script around Crack.

Foreground mode is disabled if you try running Crack-network -f on the command line, because of the insensibility of rsh'ing to several machines in turn, waiting for each one to finish before calling the next. For more information, read the section about Network Cracking without NFS/RFS in the README.NETWORK file.

-v Sets verbose mode, whereby Crack will print every guess it is trying on a per-user basis. This is a very quick way of flooding your filestore, but useful if you think something is going wrong.

-m Sends mail to any user whose password you crack by invoking Scripts/nastygram with their username as an argument. The reason for using the script is so that a degree of flexibility in the format of the mail message is supplied; i.e., you don't have to recompile code in order to change the message.

-nvalue Sets the process to be nice()ed to value, so, for example, the switch -n19 sets the Crack process to run at the lowest priority.

-network Throws Crack into network mode, in which it reads the Scripts/network.conf file, splits its input into chunks that are sized according to the power of the target machine, and calls rsh to run Crack on that machine. Options for Crack running on the target machine may be supplied on the command line (for example, verbose or recover mode), or in the network.conf file if they pertain to specific hosts (e.g., nice() values).

-r<pointfile>

This is only for use when running in recover mode. When a running Crack instance starts pass 2, it periodically saves its state in a point file, with a name of the form Runtime/P.* This file can be used to recover where you were should a host crash. Simply invoke Crack in exactly the same manner as the last time, with the addition of the **-r** switch (for example, **-rRuntime/Pfred12345**). Crack will startup and read the file, and jump to roughly where it left off. If you are cracking a very large password file, this can save a lot of time after a crash.

611-EMD-001, Rev. 02

5.2.1.5 Crack Support Scripts

The Scripts directory contains a small number of support and utility scripts, some of which are designed to help Crack users check their progress. The most useful scripts are briefly described below.

Scripts/shadmrg

This is a small script for merging /etc/passwd and /etc/shadow on System V style shadow password systems. It produces the merged data to stdout, and will need to be redirected into a file before Crack can work on it.

Scripts/plaster

This is a simple front-end to the Runtime/D* diefiles that each copy of the password cracker generates. Invoking Scripts/plaster will kill off all copies of the password cracker you are running, over the network or otherwise. Diefiles contain debugging information about the job, and are generated so that all the jobs on the entire network can be called quickly by invoking Scripts/plaster. Diefiles delete themselves after they have been run.

Scripts/status

This script rsh's to each machine mentioned in the Scripts/network.conf file, and provides some information about processes and uptime on that machine. This is useful when you want to find out just how well your password crackers are getting on during a Crack - network.

Scripts/{clean,spotless}

These are just front ends to a makefile. Invoking Scripts/clean cleans up the Crack home directory and removes unwanted files, but leaves the pre-processed dictionary bigdict intact. Scripts/spotless does the same as Scripts/clean, but obliterates bigdict and old output files, too, and compresses the feedback files into one.

Scripts/nastygram

This is the shell script that is invoked by the password cracker to send mail to users who have guessable passwords, if the **-m** option is used. Edit it to suit your system.

Scripts/guess2fbk

This script takes your out* files as arguments and reformats the 'Guessed' lines into a feedback file, suitable for storing with the others.

An occasion where this might be useful is when your cracker has guessed a large number of passwords and then died for some reason (a crash?), before writing out the guesses to a feedback file. Running **Scripts/guess2fbk out* >> Runtime/F.new** will save the work that has been done.

5.2.1.6 Checking the Log

Crack loads dictionaries directly into memory, sorts and makes them unique, before attempting to use each of the words as a guess for each users' password. If Crack correctly guesses a password, it marks the user as done and does not waste further time on trying to break that user's password.

Once Crack has finished a dictionary pass, it sweeps the list of users looking for the passwords it has cracked. It stores the cracked passwords in both plain text and encrypted forms in a feedback file in the directory **Runtime**. Feedback files have names of the form **Runtime/F***. This allows Crack to recognize passwords that it has successfully cracked previously, and filter them from the input to the password cracker. This provides an instant list of "crackable" users who have not changed their passwords since the last time Crack was run. This list appears in a file with name **out*** in the **$CRACK_OUT** directory, or on **stdout**, if foreground mode (**-f**) is invoked (see Section "Options", above).

Similarly, when a Crack run terminates normally, it writes out to the feedback file all encrypted passwords that it has NOT succeeded in cracking. Crack will then ignore all of these passwords next time you run it.

Obviously, this is not desirable if you frequently change your dictionaries or rules, so, **Scripts/mrgfbk** is provided to allow for checking the "uncrackable" passwords. This script sorts your feedback files, merges them into one, and optionally removes all traces of "uncrackable" passwords, so that your next Crack run can have a go at passwords it has not succeeded in breaking before.

mrgfbk is invoked automatically if you run **Scripts/spotless** (see Section 5.2.1.5, Crack Support Scripts, above).

5.2.2 Configuring ANLpasswd

The Argonne National Laboratory wrote ANLpasswd and has made it available to everyone as freeware. There is a simple install script that will install the components on the automount host for both SGI and Sun architectures. ANLpasswd consists of a setuid C program that is used to call the anlpasswd Perl script. The Perl script uses the Crypt::Cracklib module, which is installed with the package, a dictionary generation tool, and dictionaries that are used to match attempted passwords against possible passwords that are in the dictionary file.

It is assumed that Perl 5.6 is properly installed in /tools/perl for Sun and SGI platforms. The binary ypstuff and the anlpasswd30 script (with its soft links to anlpasswd and yppasswd) are placed in /tools/bin. The Perl includes and dictionary file should also be NFS mounted and placed in /tools/lib/anlpasswd.

Once the package is configured, the only alteration may be in the dictionary files. There are a large number of dictionary files that are included by default in this release. If there are local requirements to change them (i.e. the default has too little security or too much security), the following procedure is applicable.

1   Login to the automount host as root or **su** to root.

2   Modify the SGI /tools/lib/words directory as required (add files, modify files or remove files).

3   Remove the Sun /tools/lib/words directory contents, then copy the SGI (modified) directory to the Sun directory.

4   Login to an SGI as root or **su** to root.

5   From the SGI window, type **cd /tmp** and then press the **Return/Enter** key to change to the directory where anlpasswd-30.tar.gz is located.
   - The directory is changed to **/tmp**.

6   To explode anlpasswd-30.tar.gz, type **gzip –dc anlpasswd-30.tar.gz | tar –xovf –** and ten press the **Return/Enter** key.
   - The anlpasswd-30.tar.gz file is exploded.

7   From the SGI window, to change directory to the location of the make dictionary script, type **cd /tmp/anlpasswd/anlpasswd-3.0-sgi/cracklib25_small**, and then press the **Return/Enter** key.
   - The directory is changed to /tmp/anlpasswd/anlpasswd-3.0-sgi/cracklib25_small.

8   To run the make dictionary script, type **./makedictionary.pl** and then press the **Return/Enter** key.
   - *Note***:** perl expected to be in /tools/perl
   - The script runs.

9   From the SGI window, on completion, copy the pw_dict.* files to the automount host's /tmp directory.

10   From the automount host window, copy the /tmp/pw_dict.* files to the appropriate /tools/lib directories for ***both*** SGI and Sun architectures.

11   Logout from the automount host.

12   From the SGI window, **su** to a normal user account and check that the changes work by running /tools/bin/anlpasswd as a normal user and verify at least one of the changes and that the script still works normally (without errors).

13   From the SGI window, to delete the temp files, type **rm –rf /tmp/anlpasswd**, and then press the **Return/Enter** key.
   - The temp files are deleted.

14   Logout from the SGI.

5.2.2.1 Installing ANLpasswd

Use the procedures provided in the Release Notes for the relevant version of ANLpasswd. Release Notes are available through the "Release Notes" link at the following URL:

http://cmdm-ldo.raytheon.com/baseline/

5.2.2.2 ANLpasswd readme

The following is the README.INSTALL from the tar file with comments. This work has already been incorporated in the release. It is provided here to facilitate understanding of how the product is put together.

```
ANLpasswd is used in ECS to provide interactive password checking. It is
installed on the network in the /tools/bin directory. Local installation
is not required.

PREREQUISITES
This installation requires:
Perl 5.6.1
50Mb of disk space
It will take approximately 30 minutes to complete this installation.

INSTALLATION INSTRUCTIONS
1.  Copy the anlpasswd-30.tar.gz file to a staging area. For
convenience, /tmp is used in these instructions.

2.  Login to the automount host as root or su to root.

3.  Change directory to /tmp and explode the tarball using the commands:
```
cd /tmp ↵
gzip –dc anlpasswd30.tar.gz | tar –xovf - ↵

4. Change directory to the top level directory and run the install script using the command:
cd /tmp/anlpasswd ↵
./install_anlpasswd.pl ↵

```
This will install the /tools/bin/anlpasswd30 script with links to
/tools/bin/anlpasswd and /tools/bin/yppasswd, the dictionaries
themselves, and the dictionary indexes.

5.  If you DO have a password aging method in place, skip to step 8. If
you do not have a password aging method in place and are implementing
the password aging script, copy the
/tmp/anlpasswd/password_aging_notify.pl script to the NIS master server.
To implement this script, the following information needs to be edited
in the passwordage.pl script:

# Master NIS server

$master_host = "<NISMASTER>";

###

# Domain (used when building address to send users email)
$domain = "<DAACDOMAIN>";
```

```
#

# Location of the Shadow file
$shadow_file = "<SHADOWFILELOCATION>";

# The protected accounts - these accounts are immune to password aging


@protected_accounts = ('root');
###


# Location of the directory to backup copies of the shadow file in
$shadow_archive = "<SHADOWFILEARCHIVELOCATION>";
###

# Variables used when sendmail emails messages to users and SAs
# This to address is used when sending emails to the SAs
$to_address = "<SAADDRESSLIST>";

where:
<NISMASTER> is the fully qualified host name for the NIS master
<DAACDOMAIN> is the NIS domain name of the DAAC
<SHADOWFILELOCATION> is the location of the shadow file (normally
/etc/shadow)
<SHADOWFILEARCHIVELOCATION> is the directory of the shadow file archive
backups
< SAADDRESSLIST> is the email account(s) to send messages to SAs
```

6. Setup cron to run the script at a convenient time.

7. Logoff from the automount host.

8. Checkout the SGI installation from an SGI production host by logging in as a normal user.

9. Change your password using the command:
    % **/tools/bin/yppasswd** ↵

10. Logout

11. Wait a few minutes to make sure that the updates are completed.

12. Checkout the Solaris installation from a Sun production host by logging in as a normal user.

13. Change your password using the command:
    % **/tools/bin/yppasswd** ↵

14. Logout

15. Wait a few minutes to make sure that the updates are completed.

That should be all that is needed to get this program up and running. If there are any problems or inaccuracies in this documentation, or you have any improvements or bug fixes, please send email to "support@mcs.anl.gov"

5.3 Aging Passwords

Password aging is required by NPR 2810.1, *NASA Procedural Requirements: Security of Information Technology*. A perl script is provided as part of the ANLpasswd 3.0 release that will, after configuration, perform 120-day password aging. If your site already has a method of doing password aging, this section may be ignored. If your site does NOT have a password aging method in place and you are implementing the password aging script, copy the /tmp/anlpasswd/password_aging_notify.pl script to the NIS master server.

To implement this script, the following information needs to be edited in the password_aging_notify.pl script:

```
# Master NIS server
$master_host = "<NISMASTER>";
###

# Domain (used when building address to send users email)
$domain = "<DAACDOMAIN>";
#
# The protected accounts - these accounts are immune to password aging
@protected_accounts = ('root');

# Location of the directory to backup copies of the shadow file in
$shadow_archive = "<SHADOWFILEARCHIVELOCATION>";
###

# Variables used when sendmail emails messages to users and SAs
# This to address is used when sending emails to the SAs
$to_address = "<SAADDRESSLIST>";
```

where:
<NISMASTER> is the fully qualified host name for the NIS master
<DAACDOMAIN> is the NIS domain name of the DAAC
<SHADOWFILELOCATION> is the location of the shadow file (normally /etc/shadow)
<SHADOWFILEARCHIVELOCATION> is the directory of the shadow file archive backups
< SAADDRESSLIST> is the email account(s) to send messages to sys administrators

1 Setup cron to run the script at a convenient time.

2 Logoff from the automount host.

5.4 Secure Access through Secure Shell

The security risks involved in using "R" commands such as rlogin, rsh, rexec and rcp are well known, but their ease of use has made their use tempting in all but the most secure of environments. Ssh is an easy-to-use, drop in replacement for these commands developed by Tatu Ylonen. Ssh is a "user" level application. No changes to the host kernel are required. The UNIX server implements the commercial version of F-Secure. As of the F-Secure 3.2 release, only SSH Version 2 is included in pre-compiled, OS-specific packages

As of the Secure Shell 2.0 release in May, 2000 and later, all of the files needed to function are loaded locally on each UNIX host in /usr/local/bin.

- ssh - replaces rsh, rlogin and rexec for interactive sessions
- scp - replaces rcp for interactive file transfer
- ssh-agent – application that allows a user to enter the passphrase once, then when other applications (e.g. ssh, scp) are used, one is not prompted for the passphrase – it is automatically negotiated.
- ssh-add - add access to a specific ssh host
- ssh-keygen - generates keys for the local host based on a passphrase (long password)
- ssh-signer – verifies that a key is genuine so that public key authentication may proceed
- sftp - secure ftp

The host daemon is in /usr/local/sbin which includes:

- sshd2 - the ssh version 2 daemon

Several files are generated on installation and when running and are installed locally:

- /etc/ssh2/ssh2_config - system-wide configuration for the ssh2 client
- /etc/ssh2/hostkey - contains the long number used for one of the ssh2 keys
- /etc/ssh2/hostkey.pub - contains the ssh2 key known to the public
- /etc/ssh2/random_seed - base number used in generating keys
- /etc/ssh2/sshd2_config - defines the local ssh2 security policy
- /etc/sshd2_22.pid - the process id of the ssh2 daemon currently running

The amount of disk space that the programs and the configurations require is less than 25 MB.

5.4.1 Installation of SSH

Use the procedures provided in the Release Notes for the relevant version of ssh. Release Notes are available through the "Release Notes" link at the following URL:

http://cmdm-ldo.raytheon.com/baseline/

5.4.2 The SSH Encryption Mechanism[1]

Each host has a host-specific DSA key (normally 1024 bits) used to identify the host. Additionally, when the daemon starts, it generates a server DSA session key (normally 768 bits). This key is normally regenerated every hour if it has been used, and is never stored on disk.

Whenever a client connects the daemon, the daemon sends its host and server public keys to the client. The client compares the host key against its own database to verify that it has not changed. The client then generates a 256 bit random number. It encrypts this random number using both the host key and the server key, and sends the encrypted number to the server. Both sides then start to use this random number as a session key that is used to encrypt all further communications in the session. The rest of the session is encrypted using a conventional cipher.

---

[1] From the *sshd* man page

Under EMD the aes128 cipher is used. The client selects the encryption algorithm to use from those offered by the server.

Next, the server and the client enter an authentication dialog. The client tries to authenticate itself using .rhosts authentication, .rhosts authentication combined with DSA host authentication, RSA challenge-response authentication, or password-based authentication. (NOTE: In the EMD configuration, .rhosts is NOT available).

Rhosts authentication is disabled within the DAACs because it is fundamentally insecure.

If the client successfully authenticates itself, a dialog for preparing the session is entered. At this time the client may request things like allocating a pseudo-tty, forwarding X11 connections, forwarding TCP/IP connections, or forwarding the authentication agent connection over the secure channel.

5.4.3 How a User Uses Secure Shell

The Simplest Way

1   To login, use the command:

        % slogin defiant ↵

        Enter the passphrase for the key (lotsofstuffhere): **br0wn cow 3ats grass** ↵
        Last login: Sun Feb 22 06:50:59 1998 from echuser.east.hitc.com
        No mail.
        %

NOTE: The first time you login to a host the following message will pop up asking if you
             want to continue. In response, type **yes** and [**enter**]:

        Host key not found from the list of known hosts.
        Are you sure you want to continue connecting (yes/no)? **yes** ↵
        Host 't1acg01' added to the list of known hosts.

2   To transfer a file, use the command:

        % scp hostone:/etc/info info ↵
        Enter the passphrase for the key (lotsofstuffhere): **br0wn cow 3ats grass** ↵

        •   This will copy the file /etc/info from hostone to your local host. Note that your
             passphrase is needed to initiate the transfer.

IMPORTANT NOTE: The default directory on the *target* host is always the users
             HOME directory.

3 Also, one may send/receive files recursively using "-r" such as:

% scp -r ~/files/* hostone:~/files ↵

will send what is in the home directory files subdirectory to the target host hostone in the home files subdirectory.

4 To execute a command remotely, use the command:

% ssh whoisonfirst ps –ef ↵

Enter the passphrase for the key (lotsofstuffhere): **br0wn cow 3ats grass** ↵

5.4.4 A Layer of Convenience

If you are already a user of "r" commands, you probably know about the .rhost file. Ssh will allow a user to setup the .rhost equivalent called .shost in one's home directory. .Rhost and .shost contain the names of the hosts to which one normally connects. The nice thing about using it is one need not enter one's passphrase. Unlike "r" commands, however, ssh commands use long strings of numbers to authenticate the client, which makes it quite difficult for an intruder to impersonate a legitimate user. One word of caution, however, if you leave your terminal while logged on, a passerby could logon to any host in your .rhost/.shosts file and potentially cause malicious damage to you and your colleagues work. Be aware!

NOTE: ssh checks the mode of .shost, so change permission on .shost by typing:

% chmod 600 /home/JohnDoe/.shost ↵

where you must substitute your own home directory for /home/JohnDoe.

5.4.5 Multiple Connections

If you open multiple connections, it is more convenient to keep your keys in system memory. To do this requires executing two commands:

% ssa ↵
Enter the passphrase for the key (lotsofstuffhere):
Enter passphrase: **br0wn cow 3ats grass** ↵
Identity added: /home/JohnDoe/.ssh/identity (bpeters@nevermor)
%

Now, one may make connections (slogin, scp, ssh) to hosts that are running ssh without being prompted for a passphrase.

5.4.6 Secure FTP

A secure version of ftp is available. Use the command:

% sftp user@remotehost ↵
Enter the passphrase for the key (lotsofstuffhere): **MY PASSPHRASE** ↵
local directory - /home/user

remote directory - /home/user
sftp> get thisisafilename ↵
sftp> put thisotherfilename ↵
sftp> quit ↵

5.4.7 Other Notes

IMPORTANT: SSH will automatically "tunnel" X sessions without user involvement even through multiple hops. However, it is important that you do NOT change the DISPLAY parameter or X will not use the ssh tunnel!

5.4.8 Configuration of Secure Shell

5.4.8.1 Local Setup

Most users will start from the same host whether from an X terminal, a UNIX workstation, or a PC. Running the sss (sshsetup) script generates long strings called keys that make ssh work. One set of keys is needed for each home directory.

The only thing you need to know before executing the script is to pick a good passphrase of at least 10 characters. You can and <u>should</u> use spaces and multiple words with numbers, misspellings and special characters. Note that passwords are NOT echoed back to the screen.

PLEASE DO NOT USE THE PASSWORDS/PASSPHRASES USED HERE OR IN ANY OTHER DOCUMENTATION!

Using the script sss should look like:

% sss ↵
Use a passphrase of at least 10 characters; which should include numbers
or special characters and MAY include spaces
New passphrase: **This is a silly test** ↵
Retype new passphrase: **This is a silly test** ↵
Generating ssh1 keys. Please wait while the program completes...
Generating ssh2 keys. This can take up to 240 seconds...
Done with sshsetup!
%

You are on the way!

NOTE: If you have accounts in the PVC, VATC and/or the EDF, at a DAAC production LAN or DAAC M&O LAN, do sss in EACH environment.

5.4.8.2 Remote Setup

If you need to access a host with a different home directory, you will need to run the ssr (ssh remote) script. NOTE: It is helpful to have run Secure Shell Setup (sss) in each environment first before doing the ssh remote script. This script sets up the destination host with the new set

of keys and transfers the source (local) key to the destination and the destination key to the source. A new capability is to use different user names on the source and target hosts. This should look something like:

% **ssr** ↵
Remote user name (default: yourusername): ↵
Do you want to setup for:
1   VATC
2   PVC
3   MiniDAAC
4   GSFC DAAC
5   SMC
6   GSFC M and O
7   EDC DAAC
8   EDC M and O
9   LaRC DAAC
10   LaRC M and O
11 NSIDC DAAC
12 NSIDC M and O
x   Exit from script
Select:
2
Working...
Accepting host p0spg07.pvc.ecs.nasa.gov key without checking.
yourusername@p0spg07.pvc.ecs.nasa.gov's password:
Authentication complete. Continuing with sshremote...
Downloaded remote keys.
Uploaded local keys.
Keys concatenated.

Enter next site (press the enter-key and then x enter-key to exit)
Remote user name (default: yourusername): ↵

Do you want to setup for:
1   VATC
2   PVC
3   MiniDAAC
4   GSFC DAAC
5   SMC
6   GSFC M and O
7   EDC DAAC
8   EDC M and O
9   LaRC DAAC
10   LaRC M and O
11 NSIDC DAAC
12 NSIDC M and O

```
x   Exit from script
Select:
x <enter>
bye!
%
```

5.4.8.3 Changing your Passphrase

To change your passphrase, use the following command:

% ssp ↵

Enter old passphrase: little 1amp jumb3d <enter>
Enter a new passphrase of at least 10 characters which should include
numbers or special characters and MAY include spaces

New passphrase: **br0wn cows 3at grass** ↵
Retype new passphrase: **br0wn cows 3at grass** ↵

ssh2 key changed successfully.
Done with sshpass2!

5.4.9 Administration of Secure Shell

There is no administration of secure shell required except for general monitoring to make sure that the daemon process (/usr/local/sbin/sshd2) is running. Note, however, that the standard installation will establish a /var/log/ssh log file. It is recommended to review the /var/log/ssh and the system log file at least once a week.

5.5 Controlling Requests for Network Services (TCP Wrappers)

With TCP Wrappers, you can monitor and filter incoming requests for network services, such as FTP.

TCP Wrapper provides a small wrapper program for inet daemons that can be installed without any changes to existing software or to existing configuration files. The wrappers report the name of the client host and the name of the requested service; the wrappers do not exchange information with the client or server applications, and impose no overhead on the actual conversation between the client and server applications. The usual approach is to run one single daemon process that waits for all kinds of incoming network connections.   Whenever a connection is established, this daemon runs the appropriate server program and goes back to sleep, waiting for other connections.

Operations personnel will monitor requests for these network services:

Client	Server	Application
ftp	Ftpd	file transfer
finger	Fingerd	show users

The **/var/log/wrappers log** file should be reviewed at least once a week. The log file provides information concerning who tried to access the network service. TCP Wrapper blocks any request made by unauthorized users. TCP Wrapper can be configured to send a message to any administrator whose request is rejected.

NOTE: All DAACs, except for NSIDC, do not use TCP Wrappers.

5.5.1 Installation, Configuration, and Testing for Wrappers

The installation of TCP Wrappers is part of the Secure Shell 2.0 and later packages. As of F-Secure SSH 3.2, it is a separate package and should be installed as part of the ssh installation. (Refer to the Release Notes as described in Section 5.4.1.) The location of most of the wrappers files have been changed to /etc/wrappers. Libwrap.a is in /usr/local/lib and tcpd.h is in /usr/local/include. The installation is automatic if wrappers has been previously installed. After installation, however, the following checks should be made.

1   There are two files that provide access control for the system: /etc/hosts.allow and
        /etc/hosts.deny
   • The general format is:

   daemonlist : clientlist : script : ALLOW/DENY

   • What follows is an example of a /etc/hosts.allow file:

   ================
   sshd2,sshdfwd-X11,telnetd,ftpd: 127.0.0. : BANNERS /etc/wrappers/banners :
   ALLOW
   sshd2,sshdfwd-X11,ftpd: 155.157. : BANNERS /etc/wrappers/banners : ALLOW
   ALL: ALL: BANNERS /etc/wrappers/banners : DENY

2   Verify that the following lines are included in a Solaris /etc/inetd.conf :

   ftp   stream tcp   nowait root /etc/wrappers/tcpd   in.ftpd
   telnet stream tcp   nowait root   / etc/wrappers/tcpd   in.telnetd

   • Verify that the default daemons are commented out:

   #ftp   stream tcp   nowait root   /usr/sbin/in.ftpd   in.ftpd
   #telnet stream tcp   nowait root   /usr/sbin/in.telnetd   in.telnetd

3   The following lines should be included in an IRIX /etc/inetd.conf:

   ftp   stream tcp   nowait root /etc/wrappers/tcpd   ftpd –lll
   telnet stream tcp   nowait root   / etc/wrappers/tcpd   telnetd

   • The default daemons should be commented out:

   #ftp   stream tcp   nowait root   /usr/etc/ftpd   ftpd –lll
   #telnet stream tcp   nowait root   /usr/etc/telnetd   telnetd

4   Verify that wrappers is functioning by telnetting to the installed system from a different local host. (telnet is turned off from the outside). You should get a banner and a log entry.

5   You are done.

5.6 Monitoring File and Directory Integrity (Tripwire)

Tripwire is a tool that aids in the detection of unauthorized modification of files resident on UNIX systems. One important application of Tripwire is its use as the first and most fundamental layer of intrusion detection for an organization. Tripwire is automatically invoked at system startup. This utility will check the file and directory integrity by comparing a designated set of files and directories against information stored in a previously generated database. Tripwire flags and logs any differences, including added or deleted entries. When run against system files regularly, Tripwire spots any changes in critical system files, records these changes into its database, and notifies system administrators of corrupted or tampered files so that they can take damage control measures quickly and effectively. With Tripwire, system administrators can conclude with a high degree of certainty that a given set of files remain free of unauthorized modifications if Tripwire reports no changes. Tripwire works in conjunction with these other solutions to provide a "Defense in Depth"(trademark) security solution.

NOTE: Since system files should not change and users' files change constantly, Tripwire should be used to **monitor only system files**. The list of system files you want to monitor is stored in **./configs**/**tw.conf**.

The system administrator should install Tripwire on a clean system. This baseline database will then be used to compare possible changes to files and directories to make sure the system has not been compromised. If the system has been compromised, information provided by Tripwire can be used to carry out a forensics investigation of the compromise. Forensics is the compiling of the chain of evidence necessary to prosecute offenders after an attack has occurred.

The system administrator should check any changes made to the system on a weekly basis or after an alert from a security organization like NASIRC or CERT has put out an alert on security vulnerabilities for any of the baseline operating systems or COTS software.

All reported changes need to be investigated right away. The investigator should be aware that most of the file changes are due to system updates. But each change should be traceable to a specific, baselined change. If no unexplained changes are detected, then the Tripwire database needs to be updated to reflect file updates. Tripwire should be configured to mail the system administrator any output that it generates.

5.6.1 Installation of Tripwire

Use the procedures provided in the Release Notes for the relevant version of Tripwire. Release Notes are available through the "Release Notes" link at the following URL:

http://cmdm-ldo.raytheon.com/baseline/

5.6.2 Updating the Tripwire Database

You can update your Tripwire database in two ways. The first method is interactive, where Tripwire prompts the user whether each changed entry should be updated to reflect the current state of the file, while the second method is a command-line driven mode where specific files/entries are specified at run-time.

5.6.2.1 Updating Tripwire Database in Interactive Mode

Running Tripwire in Interactive mode is similar to the Integrity Checking mode. However, when a file or directory is encountered that has been added, deleted, or changed from what was recorded in the database, Tripwire asks the user whether the database entry should be updated.

For example, if Tripwire is run in Interactive mode and a file's timestamp changed, Tripwire will print out what it expected the file to look like, what it actually found, and then prompt the user to specify whether the file should be updated. For example:

```
/etc/hosts.equiv
    st_mtime: Wed May  5 15:30:37 1993     Wed May  5 15:24:09 1993
    st_ctime: Wed May  5 15:30:37 1993     Wed May  5 15:24:09 1993
---> File:  /etc/hosts equiv
---> Update entry?  [YN(y)n?] y ↵
```

You could answer yes or no, where a capital 'Y' or 'N' tells Tripwire to use your answer for the rest of the files. (The 'h' and '?' choices give you help and descriptions of the various inode fields.)

While this mode may be the most convenient way of keeping your database up-to-date, it requires that the user be "at the keyboard." A more conventional command-line driven interface exists, and is described next.

5.6.2.2 Updating Tripwire Database in Database Update Mode

Tripwire supports incremental updates of its database on a per-file/directory or tw.config entry basis. Tripwire stores information in the database so it can associate any file in the database with the tw.config entry that generated it when the database was created.

Therefore, if a single file has changed, you can:

> **# tripwire -update /etc/newly.installed.file** ↵

Or, if an entire set of files that made up an entry in the tw.config file changed, you can:

> **# tripwire -update /usr/local/bin/Local_Package_Dir** ↵

In either case, Tripwire regenerates the database entries for every specified file. A backup of the old database is created in the **./databases** directory.

Tripwire can handle arbitrary numbers of arguments in Database Update mode.

The script **twdb_check.pl** script is an interim mechanism to ensure database consistency. Namely, when new entries are added to the tw.config file, database entries may no longer be associated with the proper entry number. The twdb_check.pl script analyzes the database, and remaps each database entry with its proper tw.config entry.

5.6.3 Configuring the tw.config File

Edit your **tw.config** file in the **./configs** directory, or whatever filename you defined for the Tripwire configuration file, and add all the directories that contain files that you want monitored. The format of the configuration file is described in its header and in the "man" page. Pay especially close attention to the select-flags and omit-lists, which can significantly reduce the amount of uninteresting output generated by Tripwire. For example, you will probably want to omit files like mount tables that are constantly changed by the operating system.

Run Tripwire with **tripwire -initialize**. This will create a file called **tw.db_[hostname]** in the directory you specified to hold your databases (where [hostname] will be replaced with your machine hostname).

Tripwire will detect changes made to files from this point on. You ***must*** be certain that the system on which you generate the initial database is clean; however, Tripwire cannot detect unauthorized modifications that have already been made. One way to do this would be to take the machine to single-user mode, reinstall all system binaries, and run Tripwire in initialization mode before returning to multi-user operation.

This database must be moved someplace where it cannot be modified. Because data from Tripwire is only as trustworthy as its database, choose this with care. It is recommended to place all the system databases on a read-only disk (you need to be able to change the disk to writeable during initialization and updates, however), or exporting it via read-only NFS from a "secure-server." (This pathname is hardcoded into Tripwire. Any time you change the pathname to the database repository, you must recompile Tripwire. This prevents a malicious intruder from spoofing Tripwire into giving a false "okay" message.)

We also recommend that you make a hardcopy printout of the database contents right away. In the event that you become suspicious of the integrity of the database, you will be able to manually compare information against this hardcopy.

Once you have your database set up, you can run Tripwire in Integrity Checking mode by typing **tripwire** on the command line from the directory in which Tripwire has been installed.

5.7 Reporting Security Breaches

Reporting of Security breaches shall be in accordance with NPR 2810.1, *NASA Procedural Requirements: Security of Information Technology*. The specific location in the 2810 is the section on IT Security Incidents Reporting and Handling.

5.8 Initiating Recovery from Security Breaches

Recovery from Security breaches shall be in accordance with NPR 2810.1, *NASA Procedural Requirements: Security of Information Technology*. The specific location in the 2810 is the section on IT Security Incidents Reporting and Handling.

6. Network Administration

This section covers the procedures necessary for the management operations that monitor and control the system network capabilities.

Detailed procedures for tasks performed by the Network Administrator are provided in the sections that follow. The procedures assume that the administrator is authorized and has proper access privileges to perform the tasks (i.e., root).

6.1 Network Documentation

EMD Network Administration requires access to restricted documents that are posted on the EMD Baseline Information System (EBIS) Site (http://pete.hitc.com/baseline/) but are not available on the public mirror site (http://cmdm-ldo.raytheon.com/baseline/). The following restricted documents provide network documentation:

- DAAC LAN Topology   921-TD*x*-001
  (*x* = DAAC designation: G = GSFC; L = LaRC; N = NSIDC; E = LP DAAC)
- [DAAC] Hardware/Network Diagram   921-TD*x*-002
- IP Address Assignment (DAAC Hosts)   921-TD*x*-003
- IP Address Assignment (DAAC Network Hardware)   921-TD*x*-004

The documents describe and depict the network layout and inter/intra-connections necessary to understand the system. Contact EMD Configuration Management for versions relevant to an individual site.

6.2 Network Monitoring

WhatsUp **Professional 2006 Premium (WhatsUp Professional)** is a graphical network monitoring application selected to monitor critical devices and services on the Production Local Area Network (LAN) and/or additional system networks. It initiates alerts when it detects problems, and can send remote notifications by beeper, pager, and e-mail. It logs events to facilitate troubleshooting and reporting. It is implemented on Windows XP on a Personal Computer (PC) connected to the Production LAN. Chapter 7 of this document provides basic procedures for WhatsUp **Professional**. Additional information is available in 609-EMD-001, Release 7.11 Operations Tools Manual for the EMD Project and the *WhatsUp Professional – 2006 and 2006 Premium Edition User's Guide Release Notes* vendor documentation, which is available on the Internet at the following URL:

http://support.ipswitch.com/kb/WP-20051212-DM01.htm

Monitoring is also conducted via command-line interaction and site-developed scripts. An overview of network information is provided here as a basis for evaluating network status.

6.3 DAAC LAN Topology Overview

The LAN topology at each DAAC is unique. The detailed network topology for each DAAC is not presented in the student guide due to network security concerns. However, the details will be discussed during the class presentation.

In spite of the unique design of each DAAC's LAN, there are some common features.

- The Production Network at each Distributed Active Archive Center (DAAC) consists of an Ethernet Virtual Local Area Network (VLAN) supported by Ethernet 10-Mb/s, 100-Mb/s, and Gigabit Ethernet (GigE) connections and a GigE switch.
- The VLAN Ethernet switch is connected to a Portus Firewall by a Gigabit Ethernet connection.
- The firewall provides access to both the M&O Network and a VLAN outside the firewall that provides connections to external networks, such as the following services:
  - Campus network.
  - Very high performance Backbone Network Service (vBNS+).
  - Internet Protocol Network Operations Center (IPNOC) and other parts of the NASA Integrated Services Network (NISN).

System hosts within a DAAC are connected to the VLAN Ethernet switch. The switch is used to connect hosts at 10/100/1000 Mb/s. The VLAN Ethernet switch is connected to the Portus Firewall via a 1000-Mb/s (GigE) connection.

In addition to their connections to the VLAN Ethernet switch the ACM/Ingest hosts, PDR Server hosts, Data Server hosts, and science processors are interconnected through a backend GigE switched network. This setup provides a high-speed network to handle the large data transfers among subsystems. The GigE switch shifts the numerous transfers of large volumes of data onto a dedicated high-speed topology, freeing the VLAN to handle control flows and DAAC-to-DAAC processing flows.

6.4 Network Hardware Components

The DAAC LANs consist of the following major hardware components:

- Portus Firewall.
- VLAN Ethernet Switch.
- GigE Switch.

6.4.1 Portus Firewall

The Portus Firewall hardware consists of an IBM RS6000 server installed with the basic AIX 5.3operating system. It contains two 9GB internal disk drives that are mirrored, as well as a pair of redundant power supplies. Distributed FTProxy SOCKS is provided by Portus Forewall, which has replaced E-Borders Server..

6.4.2 VLAN Ethernet Switch

The Ethernet switch at each DAAC is a Cisco Catalyst 6506E , which provide a large number of 10/100/1000-Mb/s interfaces. The VLAN1 Ethernet switch interfaces with all Production hosts and the Portus Firewall. The VLAN10 Ethernet switch interfaces with the Portus Firewall and routers to external networks.

Maintenance and configuration of the Ethernet switche is considered non-trivial functions. Such tasks are addressed in special technical training provided by the vendor and supplemental training provided by EMD.

6.4.3 GigE Switch

The GigE Switch is Cisco Catalyst 3550. It connects some of the SGI hosts to a high-speed switching fabric.

Maintenance and configuration of the GigE Switch are considered non-trivial functions. Such tasks are addressed in special technical training provided by the vendor and supplemental training provided by EMD.

6.5 Domain Name Service (DNS) Structure

The parent DNS domain for the system is **ecs.nasa.gov**. These DNS servers reside at the SMC, and the NSIDC DAAC. In this domain are the SMC hosts, User hosts for all DAACs, and pointers to the DAACs' external DNS servers. The external DNS resides on the Portus Firewall.

The ecs.nasa.gov DNS servers are:

- m0isi08.ecs.nasa.gov
- n0fwi09u.ecs.nasa.gov

The DAACs' Production networks are a child domain of ecs.nasa.gov. They are:

- LaRC Production networks:
  - l0ins02.larcb.ecs.nasa.gov (internal)
  - l0css02.larcb.ecs.nasa.gov (internal)
  - l0isi08.larcb.ecs.nasa.gov (external)
- LP DAAC Production networks:
  - e0ins02.edcb.ecs.nasa.gov (internal)
  - e0css02.edcb.ecs.nasa.gov (internal)
  - e0fwi09.edcb.ecs.nasa.gov (external)
- NSIDC Production network:
  - n0ins02.nsidcb.ecs.nasa.gov (internal)
  - n0css02.nsidcb.ecs.nasa.gov (internal)
  - n0fwi09.nsidcb.ecs.nasa.gov (external)
- GSFC Production networks:
  - g0ins02.gsfcb.ecs.nasa.gov (internal)
  - g0css02.gsfcb.ecs.nasa.gov (internal)

–   g0fwi09.gsfcb.ecs.nasa.gov (external)

The DAACs' M&O networks are also a child domain of ecs.nasa.gov. They are:

- LaRC M&O network.
  - larcmo.ecs.nasa.gov
- LP DAAC M&O network.
  - edcmo.ecs.nasa.gov
- NSIDC M&O network.
  - nsidcmo.ecs.nasa.gov
- GSFC M&O network.
  - gsfcmo.ecs.nasa.gov

6.6 Host Names

A letter is appended to the production host name to distinguish which interface (and IP address) a user is accessing.

As an example, a GSFC DAAC host named g0acg01.gsfcb.ecs.nasa.gov is a host attached to the Production network.

6.7 Network Security

6.7.1 Network Connectivity

The system network was designed to minimize unauthorized user access, including the use of a firewall at each site. Ingest network access at a DAAC is limited to its Level 0 data provider(s), the SMC, and hosts attached to the DAAC's Production and M&O networks. No local campus, Internet or other DAAC access is provided. Access to a DAAC's Production network is limited to the SMC, the DAAC's M&O network, and other DAACs. No local campus, Internet, or Level 0 data provider(s) access is provided.

6.7.2 Troubleshooting - Verifying Connectivity

One of the key reasons for failure of data access and transfer is an error or problem in system connectivity. This can be caused by a myriad of glitches such as incorrect/outdated lookup tables, incorrectly assigned IP addresses, missing default route and more. Besides checking individual host/server operation with various tools such as ECS Assistant, you can use several command line entries to verify point-to-point communication between components.

There are three initial steps to help verify system connectivity:

- Ensuring connectivity is authorized (check the Network Connectivity matrix).
- Determining whether the Domain Name Service (DNS) is resolving host name and IP addresses correctly.
- Actively testing the connectivity using the ping function.

6.7.2.1 Checking Local Host Access to Another Local Host over the Network

1   To check the Domain Name Service entries (DNS) for the source host on workstation
x0xxx## at the UNIX prompt enter:

nslookup <local_host>

• The screen display will be similar to the following:

g0spg01{mblument}[204]->nslookup g0spg01
Server: g0css02.gsfcb.ecs.nasa.gov
Address: xxx.xxx.xxx.xx
Name: g0spg01.gsfcb.ecs.nasa.gov
Address: xxx.xxx.xxx.xx

2   To check the DNS entries for other host on the Production network enter:

nslookup <other host>

• The screen display will be similar to the following:

g0spg01{mblument}[201]->nslookup g0css02
Server: g0css02.gsfcb.ecs.nasa.gov
Address: xxx.xxx.xxx.xx
Name: g0css02.gsfcb.ecs.nasa.gov
Address: xxx.xxx.xxx.xx

3   To determine the host's network interface parameters enter:

netstat -i

• The **netstat -i** command will provide the following information:

g0spg01{mblument}[201]->netstat –i
Name Mtu   Network   Address   Ipkts Ierrs   Opkts Oerrs Coll
ipg0 4352 xxx.xxx.xxxg0spg01.gsfcb. 9182666   1 8103032   0   0
hip0 65280 xxx.xxx.xg0spg01h.gsfcb. 5554524   0 6776651   0   0
xpi0 4352 xxx.xxx.xxx.xxg0spg01u.ecs. 37850320   0 14109683   3   0
xpi1 0   none   none   0 0   0   0   0
et0* 1500 none   none   0 0   0   0   0
lo0 8304 loopback   localhost   314800   0 314800   0   0

4   To determine the host's network interface enter:

ifconfig <interface>

• Using **ipg0** from the **ifconfig <interface>** data as the interface parameter, **ifconfig
ipg0**, will result in the following display:

g0spg01{mblument}[203]->ifconfig ipg0
ipg0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST>
inet xxx.xxx.xxx.xx netmask 0xffffff00 broadcast xxx.xxx.xxx.xxx

5   To ping the local host to verify inter-connectivity enter:

ping <local host>

- For example:

    g0spg01{mblument}[232]->ping g0spg01
    PING g0spg01.gsfcb.ecs.nasa.gov (xxx.xxx.xxx.xx): 56 data bytes
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=0 ttl=255 time=0 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=1 ttl=255 time=0 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=2 ttl=255 time=0 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=3 ttl=255 time=0 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=4 ttl=255 time=0 ms
    ----g0spg01.gsfcb.ecs.nasa.gov PING Statistics----
    5 packets transmitted, 5 packets received, 0% packet loss
    round-trip min/avg/max = 0/0/0 ms
    g0spg01{mblument}[233]->

6   To verify inter-connectivity with other hosts on the Production network enter:

ping <remote host>

- For example:

    g0spg01{mblument}[202]->ping g0css02
    PING g0css02.gsfcb.ecs.nasa.gov (xxx.xxx.xxx.xx): 56 data bytes
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=0 ttl=255 time=2 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=1 ttl=255 time=1 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=2 ttl=255 time=1 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=3 ttl=255 time=1 ms
    64 bytes from xxx.xxx.xxx.xx: icmp_seq=4 ttl=255 time=1 ms
    ----g0css02.gsfcb.ecs.nasa.gov PING Statistics----
    5 packets transmitted, 5 packets received, 0% packet loss
    round-trip min/avg/max = 1/1/2 ms

Note: You cannot ping hosts outside of the Production network because of the Protus firewall implemetation. See your local firewall administrator for assistance when troubleshoot connectivity issues with hosts outside of the Production LAN.

7 To check the health of the interface enter:

netstat -i

- The following type of result is returned:

```
g0spg01{mblument}[218]->netstat -i
Name Mtu  Network    Address        Ipkts      Ierrs      Opkts        Oerrs  Coll
ipg0 4352  xxx.xxx.xxxg0spg01.gsfcb.   9197317    1          8113487      0      0
hip0 65280 xxx.xxx.xg0spg01h.gsfcb.    5554541    0          6776668      0      0
xpi0 4352  xxx.xxx.xxx.xxg0spg01u.ecs. 37851779   0          14109837     3      0
xpi1 0     none         none           0          0          0            0      0
et0* 1500  none         none           0          0          0            0      0
lo0  8304  loopback   localhost        325510     0          325510       0      0
```

8 Examine the output of the **netstat** command to determine whether there are any Ierrs and/or Oerrs.

- One or two errors are acceptable, 100 errors are not.
    - A lot of errors indicate an interface problem; check the syslog for any startup or logged problems from the OS.

6.7.2.2 Checking Host Communication across external networks

You cannot ping or traceroute to hosts outside of the Production network because of the firewall implemetation. See your local firewall administrator for assistance when troubleshoot connectivity issues with hosts outside of the Production network.

This page intentionally left blank.

7. System Monitoring

This chapter covers procedures for the management operations that monitor the network and server applications. Graphical tools available to monitor system status include a COTS program, **WhatsUp Professional 2006 (Premium Edition)**, three custom software programs, **ECS Health Check GUI**, **ECS Assistant/ECS Monitor** and **EcMs-Whazzup??**, and a script **EcCsIdPingServers** that permits an operator to ping all servers. These programs provide system monitors with real-time status of the system and indications of potential problem areas. Following this introduction, sections related to System Monitoring address procedures for the following functions:

- Section 7.1   Checking the Health and Status of the Network.
- Section 7.2   Monitoring and Managing Server Applications.

For each set of functions, an **Activity** Checklist table provides an overview of the tasks to be completed. The outline of the Activity Checklist is as follows:

Column one - **Order** shows the order in which tasks could be accomplished.

Column two - **Role** lists the Role/Manager/Operator responsible for performing the task.

Column three - **Task** provides a brief explanation of the task.

Column four - **Section** provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - **Complete?** is used as a checklist to keep track of which task steps have been completed.

7.1 Checking the Health and Status of the Network

WhatsUp Professional 2006 (Premium Edition), or WhatsUp Professional for short, is a graphical network monitoring application selected to monitor critical devices and services on the Production Local Area Network (LAN) and/or additional networks. It initiates alerts when it detects problems, and can send remote notifications by beeper, pager, and e-mail. It logs activities to facilitate troubleshooting and reporting. It is implemented on Windows XP on a Personal Computer (PC) connected to the Production LAN. Additional information on WhatsUp Professional is available in 609-EMD-001 (Revision2), Release 7.11 Operations Tools Manual for the EMD Project; the product's on-line Help; and the vendor's *WhatsUp Professional and Premium Edition 2006 User's Guide* available on the Internet at the following URL:

http://www.ipswitch.com/support/whatsup_professional/guides/2006/WUPUsersGuide2006.pdf

The procedures in this section assume that the installation procedure specified in Document 914-TDA-337, WhatsUp Professional 2006 Premium Maintenance Update for EOSDIS Core System (ECS) has been executed. During the installation process the installer performs the following activities:

- Installs the WhatsUp Professional application.
- Creates a network map.
- Sets up network map alert notifications.
- Sets up a WinPopup notification message.
- Sets up an SMTP e-mail notification message.
- Sets the network map polling properties.
- Sets device properties.
- Saves the map.
- Starts WhatsUp Professional polling.

Once a network has been discovered by **WhatsUp Professional,** monitoring the state of the network can begin. Monitoring includes tasks such as checking the map for color alerts that indicate problems and checking for network changes.

The **ECS Health Check GUI** indicates the status of the EcDmV0ToEcsGateway and Data Pool. It sends inventory searches to the EcDmV0ToEcsGateway/Data Pool at a specified rate and provides warnings when a failure is registered by the GUI during the current inventory search.

Table 7.1-1 provides an Activity Checklist for checking the health and status of the network.

Table 7.1-1. Checking the Health and Status of the Network -
Activity Checklist (1 of 2)

Order	Role	Task	Section	Complete?
1	System Administrator/Operations Controller	Launching WhatsUp Professional and Displaying the Network Map	(P) 7.1.1	
2	System Administrator/Operations Controller	Responding to Color Alerts and Obtaining Status of a Node	(P) 7.1.2	
3	System Administrator/Operations Controller	Configuring a Popup Menu for a Node or Multiple Nodes	(P) 7.1.3	
4	System Administrator/Operations Controller	Using the Net Tools Info Tool to Obtain Information on a Node	(P) 7.1.4.1	

Order	Role	Task	Section	Complete?
5	System Administra-tor/Operations Controller	Using the Net Tools Ping Tool to Verify Connectivity on a Node	(P) 7.1.4.2	
6	System Administra-tor/Operations Controller	Using the Net Tools Traceroute Tool to Trace a Route	(P) 7.1.4.3	
7	System Administra-tor/Operations Controller	Reviewing the State Change Report	(P) 7.1.5.1	
8	System Administra-tor/Operations Controller	Starting and Using the ECS Health Check GUI	(P) 7.1.6	

7.1.1 Launching WhatsUp Professional and Displaying the Network Map

The WhatsUp Professional application and graphical user interface (GUI) are installed and run in the Windows environment on a PC. Once the application is started and being used to monitor the network, it is typically left running at all times. This is because the application must be running in order for its monitoring activities (i.e., polling and logging) to occur. Therefore, under normal circumstances, it will seldom be necessary to launch the application because it will be running continually. However, if something causes the application to stop (e.g., a failure of its host, or an inadvertent closure of the application), it will be necessary to start it again. Table 7.1-2 presents the steps required to start the WhatsUp Professional application. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Log on to the PC where WhatsUp Professional has been installed.

2   Click on the **Start** button at the lower left corner of the PC monitor and select **All Programs → Ipswitch WhatsUp Professional 2006 Premium Edition → WhatsUp Professional Premium Edition** from menus.
- The WhatsUp Professional main window is displayed.

3   If the network map is not already open, select **File → Open** from the pull-down menu.
- The **Open Device Group** dialog box is displayed.

4 Double-click on a device group (or select the name with a single click and then click on the **OK** button).
- A device view or map view of the devices in the group is displayed.

Table 7.1-2. Launching WhatsUp Professional and Displaying a Network Map

Step	What to Do	Action to Take
1	Log on to the PC where WhatsUp Professional has been installed	
2	**Start → All Programs → Ipswitch WhatsUp Professional 2006 Premium Edition → WhatsUp Professional Premium Edition**	**single-click**
3	**File→Open . . .**	**single-click**
4	**<name of a device group>**	**double-click** or **click** on the name and then on the **OK** button

7.1.2 Responding to Color Alerts and Obtaining Status of a Node

Objects that have an abnormal condition can be identified by a change in appearance on the network map. Colors may be changed, but the following default conventions apply in a map window to indicate the status of a device or service:

- Device name highlighted: indicates that WhatsUp Professional has recorded an activity for the device in a log.
- Device icon on a green square background: indicates that the device is up (i.e., responds to polling).
- Device icon on a light green diamond-shaped background: indicates that the device has missed at least one polling request.
- Device icon on a yellow diamond-shaped background: indicates that the device has missed two polling requests.
- Device icon on a red elongated diamond-shaped background: indicates that the device is down (i.e., is not accessible or has missed four consecutive polling requests). Once the device has missed eight polling requests, the background is changed to a dark red starburst.
- Device icon on a light purple octagon-shaped background: indicates that a standard service on the device is down.
- Device on a gray square background: indicates monitoring has been turned off for the device.

A color alert on a symbol indicates that some part of that object may have problems. Placing your mouse's cursor on the symbol causes a brief description of the problem to appear on the status line at the bottom of the display. To help isolate a fault on the network further, it is possible to click with the right (or non-preferred) mouse button on the symbol with the color alert and bring up a variety of reports that provide more detailed information about the node

based on TCP/IP polling, the Internet Control Message Protocol (ICMP) status, and the status of services on the node.

Table 7.1-3 presents the steps required to respond to color alerts and obtain the status of a node. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   With the network map open, use the right (or non-preferred) mouse button to click on the icon for the node showing a color alert (i.e., the node label is highlighted if there has been an entry in the Activity Log related to the alert and the background is other than a green square or whatever you have selected as the indication for normal status).
- A popup menu is displayed.

2   On the popup menu, click on **<u>Device Reports . . .</u>**.
- A listing of available reports such as Device Status, Health, and State Change Timeline is displayed grouped by category. Click on the name of any report to view it.   When done reviewing the report, click the Map View tab at the bottom of the display to return to the network map or use the .

3   To acknowledge the alert, follow menu path **Tools→Acknowledge**.
- The highlighting is removed from the node label and additional instances of the alert on the node are prevented (unless the alert has been configured to be sent regardless of the acknowledgement – see **User's Guide** and on-line **Help**).

Table 7.1-3. Responding to Color Alerts and Obtaining the Status of a Node

Step	What to Do	Action to Take
1	Display the popup menu for a node with an alert	**(non-preferred) click**
2	Open **Device Reports…** dialog	**single-click**
3	Review status and related information	**read text, interpret graphs, click(s)**
4	to Click on **Map View** to return to the network map.	**single-click**
5	**Tools→Acknowledge**	**single-click**

7.1.3 Configuring a Popup Menu for a Node

The popup menu accessible when clicking the right (or non-preferred) mouse on a network map typically includes the following choices beyond the customary Copy, Paste, and Delete:
- **Display** – control the appearance of the display
- **Attach –** draw an attached line between two devices to show an arbitrary connection between devices. These are visual representations assigned by the user, and not reflections of true connectivity between the two devices; that is done with link lines.

- **Link** – set up connecting link lines on a map. Use Link lines to monitor the status of the network link (the Interface service) between two devices. Link lines can also show the status of any service which has an Active Monitor on the device.
- **Set Dependencies –** either "Set *Up Dependency* on or Set *Down Dependency* on to avoid turning off polling to devices, and instead make polling dependent on the status of another device in your database
- **Lock Position –** use Lock Position to keep an object from moving as you move other items around, or when adding devices to the map. If you want an object to be able to change positions on the map, remove the "lock position" selection.
- **Poll Now** – initiate a single poll of the network.
- **Acknowledge –** acknowledge the changed state of a device. This removes the device from Acknowledgment mode, which displays the name of the device in bold in the Device List and in a black background in Map View.
- **Device Reports. . .** – open the **Device Reports** dialog box to choose from a list of reports to display about the device.
- **Group Reports. . .** – open the **Group Reports** dialog box to choose from a list of reports to display about the device group.
- **4 Traceroute** – start the Traceroute tool to examine the network path and the intervening routers from the WhatsUp Professional machine to the device.
- **3 Ping** – start the Ping tool to send ICMP packets to the device and view the results.
- **2 Connect** – open a telnet session on the device represented by the node on the map.
- **1 Browse** – start the default browser using the IP address as the URL.
- **Remote Desktop. . .** – establish a connection with a terminal server or another computer running Windows.
- **Device Status Report. . . --** display the **Device Status Report** for the device.
- **Properties . . .** – open the **Device Properties** dialog box to permit setting parameters for the device, including General functions, monitoring functions, services, alerts, and other categories (see **User's Guide** or on-line **Help**).

The popup menu may be configured or customized for an individual device using the **Device Properties** dialog box. This will override – for the selected device – any context menu already created for the device type. Table 7.1-4 presents the steps required to configure the popup menu for a node. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   On the network map, select the node for which the popup menu is to be configured.

2   Use the right (or non-preferred) mouse button to click on the selected node.
- A popup menu is displayed.

3   On the popup menu, click on **Properties . . .**.
- The **Device Properties** dialog box is displayed to permit customization of the menu for that node.

- Click on the **Customize the menu on this device (don't use device type menu** field to set the check box. This enables the **Add . . ., Edit . . ., and Remove . . .** buttons.

4   To add a menu item, click on the **Add** button.
- The **Edit Menu Item** dialog box is displayed with three empty fields: (1) **Display name:** (2) **Command:** and (3) **Arguments:**. Using this box, it is possible to create a menu item for starting a program when the item is chosen. The Display name: field is used to specify the name of the menu item that will appear in the popup menu. The Command: field is used to enter the (file)name of any executable program to be started when the menu item is chosen from the popup menu. The Arguments: field is used to pass parameters to the specified program. See the **User's Guide** or on-line **Help** for detailed information on establishing and using popup menu items to run programs.

5   To select a displayed menu item for editing or moving click on the menu item in the list.
- The selected item is highlighted.

6   To edit a selected item, click on the **Edit** button.
- The **Edit Menu Item** dialog box is displayed as in Step 4, with information for the selected item displayed in its three fields. The displayed data may be edited to change the menu display and/or actions (see **User's Guide** or on-line **Help**).

7   To delete a selected itemclick on the **Remove** button.
- A confirmation dialog is displayed to ensure that you would like to remove the item; click on the **Yes** button to confirm.

8   Click on the **OK** button.
- The menu changes are applied and the **Device Properties** dialog is closed.

Table 7.1-4. Configuring the Popup Menu for a Node

Step	What to Do	Action to Take
1	Select the node for which the popup menu is to be configured	**single-click**
2	Display popup menu	**(non-preferred) click**
3	Open **Properties**	**single-click**
4	Set "**Customize menu…** field	**single click**
5	To add a menu item, activate the **Add** button	**single-click**
6	To select a displayed item for editing or moving, highlight the item in the list	**single-click**
7	To edit a selected item, activate the **Edit** button	**single-click**
7	To delete a selected item for a single node, activate the **Remove** button	**single-click**
8	Activate the **OK** button	**single-click**

7.1.4   Using Network Tools

WhatsUp Professional provides a set of tools to display a variety of information about nodes on the network. These tools are displayed on tabs, with the parameters and results area for one tool on each tab. The tools include:

- **Info** – display a summary of device information.
- **Time** – synchronize your computer's clock with a remote time server.
- **HTML** – query a web address.
- **Ping** – verify connectivity to a host.
- **TraceRoute** – Trace and view the route to an Internet host.
- **Lookup** – query Internet domain name servers for information about hosts and name servers.
- **Finger** – display information about users on a host.
- **Whois** – display information from the network information center about Internet domain ownership and Internet groups.
- **LDAP** – (Lightweight Directory Access Protocol); search directories for names and information stored in an LDAP directory on another computer.
- **Quote** – view quotations from a quote server.
- **Scan** – scan a range of IP addresses to create a network map.
- **SNMP** – view and graph Simple Network Management Protocol (SNMP) values for a device.
- **WinNet** – View Windows Network domains, hosts, and workstations.
- **Throughput** – test data throughput on the connection between your computer and a remote computer.
- **System Info** – view information about your local system.

Not all of these tools will necessarily be appropriate for DAAC use, but the **User's Guide** and onl-line **Help** provide detailed information on all of them. Procedures for three of these useful tools are provided here.

7.1.4.1   Using the Net Tools Info Tool to Obtain Information about a Node

The **Info** tool displays a summary of information about a network host or device, including the official host name, IP address, and contact information. An Info request on a host name also pings the host to verify connectivity. Table 7.1-5 presents the steps required to use the Net Tools Info Tool to obtain information on a node. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Follow menu path **Tools→Net Tools. . .**.
- The **Net Tools** window is displayed.

2   If necessary, click on the **Info** tab to access the **Info** tab display (when the **Net Tools** window is opened, WhatsUp Professional displays the tab most recently accessed).
- The **Info** tab controls and fields are displayed.

3   In the **Host Name or IP Address:** field, type the name or IP address of the host to be queried (this must be a fully qualified host name or address).

- The typed entry is displayed in the field.

4   Click on the **Start** button.

- A **Searching . . .** indicator appears and the **Start** button toggles to **Stop** to show that the query is in progress. At any time during the query, a click on the **Stop** button stops the query.
- The results of the query are displayed. (A click on the **Clear** button erases the results from the display window.) The **List View/Report View** button permits toggling between the Report View and the List View of the results. The Report View is a summary showing:
  - Official Name.
  - Domain Name.
  - Date the record was created.
  - Date the record was last updated.
  - Date the database was last updated.
  - Contact information (from the Whois database).
  - IP Addresses and Domain Servers.
  - The List View is a detailed list of the obtained information, including the results of the ping and more extensive information on the query.

Table 7.1-5. Using the Net Tools Info Tool to Obtain Information about a Node

Step	What to Do	Action to Take
1	Follow menu path **Tools→Net Tools**	**clicks**
2	Ensure **Info** tab is displayed	**single-click**
3	In the **Host Name or IP Address:** field, type the name or IP address of host to be queried	**enter text**
4	To start the query, activate the **Start** button	**single-click**

7.1.4.2 Using the Net Tools Ping Tool to Verify Connectivity for a Node

The **Ping** tool is a network diagnostic tool used to verify connectivity to a selected system on the network. This tool sends a data packet (an ICMP "echo request") to a remote host and displays the results for each "echo reply." This pinging command also displays the time for a response to arrive in milliseconds, as well as debugging information about the network interface. Multiple instances of the **Ping** tool may be active simultaneously.

The use of the **Ping** tool provides a quick way to verify that a device is not functioning. If the ping operations do not produce any responses or they time out, then the node is probably down or otherwise unreachable over the network. See Section 7.1.5.1, Reviewing the State Change

Timeline Report, to verify activity status of the node. If a Fault has occurred see Section 8 on Problem Management and Section 21 on COTS Hardware Maintenance.

Table 7.1-6 presents the steps required to use the Net Tools Ping Tool to verify connectivity on a node. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Follow menu path **Tools→Net Tools. . .**.
- The **Net Tools** window is displayed.

2   If necessary, click on the **Ping** tab to access the **Ping** tab display (when the **Net Tools** window is opened, WhatsUp Professional displays the tab most recently accessed).
- The **Ping** tab controls and fields are displayed.

3   In the **Host Name or IP Address:** field, type the name or IP address of the host to be checked (this must be a fully qualified host name or address).
- The typed entry is displayed in the field.

4   If it is desired to change the default number of pings to be sent, click at the end of the **Count:** field.
- The cursor is displayed at the end of the **Count:** field.

5   To set a new value for **Count:** use the **Backspace** key to remove the current value, and type the new value.
- The typed value appears in the **Count:** field.

6   Repeat Steps 5 and 6 for other options you wish to change, substituting **Delay (sec.):**, **Size**, or **Timeout (ms):** for the field name of the option to be changed, specifying respectively the number of seconds to wait between pings, the length in bytes of each packet to be sent by the **Ping** command, and the number of milliseconds of non-response from the host to be considered a failure of the ping.

7   Click on the **Start** button.
- The **Start** button toggles to **Stop** to show that the ping operation is in progress. At any time during the operation, a click on the **Stop** button stops the pinging.
- The display field at the bottom of the window shows the results of the pings. (A click on the **Clear** button erases the results from the display window.) The **List View/Report View** button permits toggling between the Report View and the List View of the results. The Report View provides, for each ping as it occurs, the address, the number of bytes sent, the response time, and the status. The List View lists the pings, the result for each packet, and the retry code.

Table 7.1-6. Using the Net Tools Ping Tool to Verify Connectivity for a Node

Step	What to Do	Action to Take
1	Follow menu path **Tools→Net Tools**	**clicks**
2	Ensure **Ping** tab is displayed	**single-click**
3	In the **Host Name or IP Address:** field, type the name or IP address of host to be checked	**enter text**
4	If it is desired to change the number of pings to be sent, move the cursor to the **Count:** field	**single-click**
5	To set a new value for **Count:** type the new value in the field	**enter text**
6	Repeat Steps 5 and 6 for other options to be changed	
7	To initiate the ping(s), activate the **Start** button	**single-click**

7.1.4.3 Using the Net Tools Traceroute Tool to Trace a Route

The **Traceroute** tool permits the operator to trace and view the route an IP packet follows from the local host to another host on the network. Response times are displayed in milliseconds and vary depending on network load. **Traceroute** can be helpful for finding potential trouble spots on large and complex networks that are connected by routers. The results of a traceroute operation can be displayed on a network map.

Table 7.1-7 presents the steps required to use the Net Tools Traceroute Tool to trace a route. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Follow menu path **Tools→Net Tools. . .**.
   - The **Net Tools** window is displayed.

2   If necessary, click on the **Traceroute** tab to access the **Traceroute** tab display (when the **Net Tools** window is opened, WhatsUp Professional displays the tab most recently accessed).
   - The **Traceroute** tab controls and fields are displayed.

3   In the **Host Name or IP Address:** field, type the name or IP address of the host to which the route is to be traced (this must be a fully qualified host name or address).
   - The typed entry is displayed in the field.

4   If it is desired to change the maximum number of hops to trace before ending the traceroute operation (a "hop" is the passing of an IP packet from one host to another), click at the end of the **Maximum Hopcount:** field.
   - The cursor is displayed at the end of the **Maximum Hopcount:** field.

5   To set a new value for **Maximum Hopcount:** use the **Backspace** key to remove the current value, and type the new value.

- The typed value appears in the **Maximum Hopcount:** field.

6   If it is desired to change the number of milliseconds of non-response from the host to cause the Traceroute to fail, click at the end of the **Timeout (ms):** field.

- The cursor is displayed at the end of the **Timeout (ms):** field.

7   To set a new value for **Timeout (ms):** use the **Backspace** key to remove the current value, and type the new value.

- The typed value appears in the **Timeout (ms):** field.

8   If it is desired to specify that the host names of each router along the route be displayed along with the IP addresses, click on the **Resolve Addresses** checkbox.

- The clicked box displays a checkmark to indicate its selection, and when **Traceroute** is run, the host names as well as the IP addresses will be shown for each router (instead of just the IP addresses). This will add time to the Traceroute operation to resolve the IP addresses.

9   Click on the **Start** button.

- An indicator shows the Traceroute operation in progress and the **Start** button toggles to **Stop** to show that the operation is in progress. At any time during the operation, a click on the **Stop** button stops the tracing.
- The display field at the bottom of the window shows the results of the traceroute operation. (A click on the **Clear** button erases the results from the display window.) The **List View/Report View** button permits toggling between the Report View and the List View of the results. The Report View provides for each hop as it occurs the address, the response time or Round Trip Time (RTT), and the status. The List View lists the hops, addresses, and more detailed information on the tracing of the route.

Table 7.1-7. Using the Net Tools Traceroute Tool to Trace a Route

Step	What to Do	Action to Take
1	Follow menu path **Tools→Net Tools**	**clicks**
2	Ensure **Traceroute** tab is displayed	**single-click**
3	In the **Host Name or IP Address:** field, type the name or IP address of host to which the route is to be traced	**enter text**
4	If it is desired to change the maximum number of hops to trace before ending the traceroute operation, move the cursor to the **Maximum Hopcount:** field	**single-click**
5	To set a new value for **Maximum Hopcount:** type the new value in the field	**enter text**
6	If it is desired to change the number of milliseconds of non-response from the host to cause the Traceroute to fail, move the cursor to the **Timeout (ms):** field	**single-click**
7	To set a new value for **Timeout (ms):** type the new value in the field	**enter text**
8	If router names are desired in the Traceroute results, select **Resolve Addresses**	**single-click**
9	To initiate the Traceroute operation, activate the **Start** button	**single-click**

7.1.5 Using WhatsUp Professional Reports

WhatsUp Professional uses a single database to store all device information across all device groups. This information is accessible through a variety of reports. Details about the nature of the logged data is provided in the **User's Guide** and on-line **Help**. The **User's Guide** and on-line **Help** also describe how to change the way events are logged, and how to generate reports to show the status of the network from various perspectives. Only the State Change Timeline report is described here, because of its potential utility in troubleshooting.

7.1.5.1 Reviewing the State Change Timeline Report

The State Change Timeline report presents a timeline of when each Monitor on a device in the selected group changed from one state to another during the displayed time period For example, if the network map shows a color alert for a device (see Section 7.1.2) and the device does not respond to a ping (see Section 7.1.4.2), the State Change Timeline will provide additional information concerning when and how the device went "down".

Table 7.1-8 presents the steps required for reviewing the State Change Timeline report. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Follow menu path **Reports→Group . . .**.

- The **Group Reports** window is displayed. Click on **State Change Timeline** from the listing of Historical Availability reports. A list is displayed describing state change events recorded by monitors for devices during the selected timeframe. *Note*: The default date range is the current day (today). Column headers can be clicked to sort the list by Start Time, Device-Monitor, State, Duration, and Message.

2   Review the list of events to identify, for example, when a monitor reported a device as Down.

- The report provides the date and time of the event and its duration and for Down events, the reason why the event occurred.
- Click on an entry in the Device-Monitor column to obtain a filtered list containing only the events for the selected device.

3   To view activities for a different time period, click on the **Back** icon (⇐), then select a new ***<date range>*** and click **GO**.

- The activities from the selected timeframe are displayed.

4   To print the contents of the display, either select **File -> Print** from the WhatsUp toolbar, or click the **Printer friendly version** icon at the tip-right corner of the report. Clicking the icon opens the report in your system's default web browser for printing.

5   To export the report in text or Excel format, open the Export Report dialog box by clicking the **Export Report** icon. Choose whether to **Export to text** or **Export to Excel** and whether to include the title and/or column names. Also choose a **column delimiter** and a **text qualifier**. Then click **Export**, which displays the File Download dialog box. Here choose whether to **Open** the file, **Save** it, or **Cancel** the export.

Table 7.1-8. Reviewing the State Change Timeline Report

Step	What to Do	Action to Take
1	Follow menu path **Reports→Group…**, and run the State Change Timeline report	**clicks**
2	Review list of events for suspect device	**read text**
3	To view activities from other than the current day, specify a new ***<date range>***	**enter date-range and click**
4	To print the contents of the display, select **File→Print** and, in the resulting dialog, specify print options; or, activate the **Printer friendly version** icon and print from your web browser.	**clicks**
5	To export the contents of the display, activate the **Export Report** icon, select options, click on **Export**, then click to **Open**, **Save** or **Cancel**	**clicks**

7.1.6 Starting and Using the ECS Health Check GUI

The **ECS Health Check GUI** indicates the status of the EcDmV0ToEcsGateway and Data Pool drill-down search engine. At a specified rate it sends inventory searches to the EcDmV0ToEcsGateway and/or submits granule search requests to the Data Pool. It provides warnings by the following means when a failure is registered by the GUI during the current search:

- Visual warning (including details about the time and nature of the error).
- Audible alarm (when implemented).
- E-mail message.

Table 7.1-9 presents the steps required for starting and using the ECS Health Check GUI. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Log in to the Sun Consolidation Internal Server host as cmshared.
- The EcCsHealthCheckStart script must be run by cmshared.
- Examples of Sun Consolidation Internal Server hosts include **g0acs11**, **e0acs11**, **l0acs03**, and **n0acs04**.
- For detailed instructions refer to the **Log in to ECS** procedure (previous section of this lesson).

2   At the UNIX prompt type **cd /usr/ecs/<*MODE*>/CUSTOM/bin/CSS** and then press the **Return/Enter** key.
- Change to the directory containing the start script (i.e., EcCsHealthCheckStart) for the ECS Health Check GUI.
- The location is different from that of most other ECS start scripts because the COTS dependency on the V0ImsClientSimulator application requires that the parent application be run in the same directory as the V0ImsClientSimulator binary.

3   At the UNIX prompt type **EcCsHealthCheckStart <*MODE*>** and then press the **Return/Enter** key.
- The ECS Health Check GUI is displayed.

4   Click on the appropriate tab for the area to be monitored.
- The following choices are available:
  - **EcDmV0ToEcsGateway.**
  - **ECS Datapools.**

5   In the **Specify timeout period minutes** and **seconds** text entry boxes type the number of minutes and seconds (respectively) before timeout.
- **Timeout period** specifies how long the GUI will wait for a response from the current search before flagging an error.

6   In the **Specify repeat period minutes** and **seconds** text entry boxes type the number of minutes and seconds (respectively) before repeating a search.

- **Repeat period** specifies how often the GUI sends an inventory search to the area being monitored.

7   Click on the **Start** button in the **Control** pane to start checking the selected area.

- The ECS Health Check GUI starts sending inventory searches to the selected area at the frequency specified in the **Specify repeat period** text boxes.
- **The Current Status is:** (as displayed on the GUI) changes from **Dormant** to **Running**.

8   Repeat Steps 4 through 7 to set up inventory searches of the other area (if applicable).

9   Observe information displayed in the **Visual Warnings** pane of the ECS Health Check GUI and listen for audible warnings (if enabled).

- Nothing much will appear to happen unless an error occurs in an inventory search sent to the EcDmV0ToEcsGateway/Data Pool, in which case the following indications will be evident:
    - **The Current Status is:** (as displayed on the GUI) changes from **Running** to **Failed** (in red).
    - Details concerning the time and nature of the error are displayed in the **Visual Warnings** text pane.
    - The **Mail Warnings** pane indicates that mail has been sent successfully to the recipients in the list.
    - A repeating audible alarm sounds (if enabled).
- If an error occurs in a search sent to the EcDmV0ToEcsGateway/Data Pool, no further inventory searches will be sent to the selected area unless restarted using the **Start** button in the **Control** pane (refer to Step 13).

10   To stop an audible alarm (when applicable) click on the **Stop** button in the **Audible Warnings** pane.

- The alarm stops sounding.

11   To clear error information from the ECS Health Check GUI (when applicable) click on the **Reset** button in the **Control** pane.

- The error information is cleared from the ECS Health Check GUI.

12   To stop inventory searches of the selected area (when applicable) click on the **Stop** button in the **Control** pane.

- The GUI goes to a dormant state.

13   To restart inventory searches of the selected area (when applicable) return to Step 7 (click on the **Start** button in the **Control** pane).

- The GUI goes to a dormant state.

14 To exit from the ECS Health Check GUI (when applicable) select **File** → **Exit** from the pull-down menu.

- The ECS Health Check GUI is dismissed.

Table 7.1-9. Starting and Using the ECS Health Check GUI

Step	What to Do	Action to Take
1	Log in to the Sun Consolidation Internal Server host	
2	**cd /usr/ecs/<*MODE*>/CUSTOM/bin/CSS**	**enter text; press Return/Enter**
3	**EcCsHealthCheckStart <*MODE*>**	**enter text; press Return/Enter**
4	**EcDmV0ToEcsGateway** or **ECS Datapools** tab (as applicable)	**single-click**
5	*minutes/seconds* (in the **Specify timeout period minutes** and **seconds** text entry boxes)	**enter text**
6	*minutes/seconds* (in the **Specify repeat period minutes** and **seconds** text entry boxes)	**enter text**
7	**Start** button (in the **Control** pane)	**single-click**
8	Repeat Steps 4 through 7 (if applicable).	
9	Observe information displayed in the **Visual Warnings** pane of the ECS Health Check GUI and listen for audible warnings (if enabled)	**read text; listen for sound**
10	**Stop** button (in the **Audible Warnings** pane) (when applicable)	**single-click**
11	**Reset** button (in the **Control** pane) (when applicable)	**single-click**
12	**Stop button (in the Control pane) (when applicable)**	**single-click**
13	**Return to Step 7 (to restart inventory searches of the selected area) (when applicable)**	
14	**File** → **Exit (when applicable)**	**single-click**

7.2 Monitoring and Managing Server Applications

There are two applications and an accompanying script provided for monitoring and managing server applications. **Whazzup???** is a management tool that provides operators and maintainers with a means of monitoring and checking servers, for quickly identifying servers that may have problems, and for isolating faults. It is a web-based application, and is therefore accessed by means of browser software. It provides the following general features:

- Host and mode views of network resources.
- Status information on resources (indicated by color coding: purple indicates inability to ping the specified host, blue indicates incomplete data collection, red indicates that the server is down, and yellow indicates that a warning threshold has been exceeded).
- Performance monitoring capability.

Another set of tools for monitoring and managing system resources is **ECS Assistant** and its companion, **ECS Monitor**, which offer:

- Installation support.
- Indication of network and server status and changes.

There is an accompanying script, **EcCsIdPingServers**, which provides the capability to ping all servers.

Table 7.2-1 provides an Activity Checklist for monitoring and managing server applications.

Table 7.2-1. Monitoring and Managing Server Applications - Activity Checklist

Order	Role	Task	Section	Complete?
1	System Administra-tor/Operations Controller	Launching EcMs-Whazzup?? and Determining What's Down	(P) 7.2.1	
2	System Administra-tor/Operations Controller	Starting ECS Assistant	(P) 7.2.2.1	
3	System Administra-tor/Operations Controller	Starting ECS Monitor	(P) 7.2.2.2	
4	System Administra-tor/Operations Controller	Using EcCsIdPingServers to Ping All Servers in a Mode	(P) 7.2.2.3	

7.2.1 Launching EcMs-Whazzup?? and Determining What's Down

A powerful COTS program that has been modified for monitoring the system is EcMsWz-Whazzup??. It is a web-accessed program that provides a graphical display of Host Status, Mode Status, Mode Verification and Performance Management. The welcome screen has buttons and links at the bottom permitting an operator to view status by various means (e.g., host, mode), verify modes and view what servers may be down, and access data on performance. The **Performance Stats** screen provides a quick overview of the system status; if Whazzup is unable to ping a host, the row for that host is highlighted in purple.

These functions of Whazzup?? provide graphical displays of host and software-server status in real-time mode. When used in conjunction with WhatsUp Professional and ECS Assistant, Whazzup?? can provide System Administrators with a comprehensive knowledge of the system's status.

Table 7.2-2 presents the steps required for launching Whazzup?? and determining what's down. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   On workstation **x0ins02**, at the UNIX prompt in a terminal window, type **netscape &** and then press the **Return/Enter** key.
   - **NOTE:** The **x** in the workstation name will be a letter designating your site: **g** = GSFC, **m** = SMC, **l** = LaRC, **e** = LP DAAC, **n** = NSIDC (e.g., **l0ins02** indicates an interface host workstation at LaRC).

2   In the location field, type **http://x0ins02u:5150** and press the **Return/Enter** key.
   - The **EcMsWz-Whazzup???** screen is displayed.

3   At the bottom of the screen, click on the **Verify Mode** option button and, in the resulting pop-up menu, drag the cursor to highlight the option **What's Down**.
   - The screen displays a table showing **Required Servers Currently Down . . .**, listing by mode the servers that are down.

4   Move the mouse to position the cursor on the **Performance** link, click the **Right Mouse Button**, and select **Open Link in New Window**.
   - The **Performance Stats** screen is displayed in a new window, showing information that may help determine the reason for any servers being down.

5   If desired, click on the link for any host to obtain more detailed information.
   - An information screen for the selected host is displayed, showing data on system memory, disk utilization, process information, and network information.

Table 7.2-2. Launching Whazzup?? and Determining What's Down

Step	What to Do	Action to Take
1	At the UNIX prompt, enter **netscape &**	**enter text; press Return/Enter**
2	Enter **http://x0ins02u:5150** in the location field	**enter text; press Return/Enter**
3	Use the **Verify Mode** option button to select **What's Down**	**click-drag**
4	Use right (or non-preferred) mouse button and the **Performance** link to open the **Performance Stats** screen in a new window	**(non-preferred) click; click**
5	If desired, use the link for any host to display more detailed information	**click**

7.2.2 ECS Assistant and ECS Monitor

The Whazzup tool provides a quick look capability to note whether any servers are down. The ECS Assistant and ECS Monitor tools provide additional easy-to-use tools that offer a server monitoring capability (ECS Monitor) as well as a capability to start and stop servers (ECS Assistant).

7.2.2.1 Starting ECS Assistant

Table 7.2-3 presents the steps required for starting ECS Assistant. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Log in to one of the host machines.

2   At the UNIX prompt on the host from which the ECS Assistant is to be run, type **setenv ECS_HOME /usr/ecs**, and then press the **Return/Enter** key.
- To verify the setting, type **echo $ECS_HOME**, and then press the **Return/Enter** key.

3   At the UNIX prompt, type **cd /tools/common/ea**, and then press the **Return/Enter** key.
- The working directory is changed to **/tools/common/ea**, the path where ECS Assistant is installed, and also where EcCoScriptlib may be found.

4   Type **EcCoAssist /tools/common/ea &**, and then press the **Return/Enter** key.
- The **ECS Assistant** GUI is displayed.

5   At the ECS Assistant GUI, click the **Subsystem Manager** pushbutton.
- The **Subsystem Manager** GUI is displayed.

6   Select a mode by clicking on the down arrow at the right end of the **Mode** field and then on the desired mode name in the resulting list.
- The selected mode is displayed in the **Mode** field and colored indicators show the installation status for components in that mode on the host; the legend for the color indications is at the lower right on the Subsystem Manager window.

7   In the list of subsystems, double click on the name of the subsystem of interest.
- One or more component groups appear below the selected subsystem name.

8   Double click on the name of a component group.
- One or more application groups appear below the selected component group name.

9   Double click on the name of the application group of interest.
- The applications or servers in the selected group are listed below the name of the group.

10   Single click on the name of an application or server of interest.
- The selected application or server is highlighted.
- Detailed installation information is displayed in the **Installation Statistics** window.

Table 7.2-3. Starting ECS Assistant

Step	What to Do	Action to Take
1	Log into one of the host machines	
2	**setenv ECS_HOME /usr/ecs**	**enter text; press Return/Enter**
3	**cd /tools/common/ea**	**enter text; press Return/Enter**
4	**EcCoAssist /tools/common/ea &** (starts the GUI)	**enter text; press Return/Enter**
5	Activate the **Subsystem Manager** pushbutton	**single-click**
6	Use the down arrow at the right end of the **Mode** field to select the desired mode	**clicks**
7	From the listed subsystems, display the component groups for the subsystem of interest	**double-click** subsystem name
8	From the listed component groups, display the application groups for any component group of interest	**double-click component group name**
9	From the listed application groups, display the applications or servers for any application group of interest	**double-click application group name**
10	From the list of applications or servers, select an application or server and display detailed information concerning its installation	**single-click application or server name**

7.2.2.2 Starting ECS Monitor

ECS Monitor provides a convenient way to monitor the status of the servers by listing their up/down condition. The **ECS Monitor** GUI has a status flag for a server indicating whether or not that server is running, and for a server that is running, the window shows the process ID (PID), the user ID, and the start time.

Table 7.2-4 presents the steps required for starting ECS Monitor. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Log in to one of the host machines.

2   At the UNIX prompt on the host from which the ECS Assistant is to be run, type **setenv ECS_HOME /usr/ecs**, and then press the **Return/Enter** key.
- To verify the setting, type **echo $ECS_HOME**, and then press the **Return/Enter** key.

3   At the UNIX prompt, type **cd /tools/common/ea**, and then press the **Return/Enter** key.
- The working directory is changed to **/tools/common/ea**, the path where ECS Monitor is installed, and also where EcCoScriptlib may be found.

4   Type **EcCoMonitorGui /tools/common/ea *<MODE>* &**, and then press the **Return/Enter** key.
- The **ECS Monitor GUI** is displayed, showing the status (**UP** or **DOWN**) of the servers on the current host in the mode specified in the command, as indicated near the top left corner of the window.
- The status "**UP/DOWN**" indicates whether a listed server is running.

5   To update the **Server Monitor** to show the current status at any time, click on the **update** button in the GUI.
- This causes the list to update to the current status.

6   To monitor other servers, log in to other hosts and launch the ECS Monitor GUI in the desired mode, as in Steps 1 through 4.

7   To exit click the **EXIT** button.
- This terminates display of the ECS Monitor GUI.

Table 7.2-4. Starting ECS Monitor

Step	What to Do	Action to Take
1	Log into one of the host machines	
2	**setenv ECS_HOME /usr/ecs**	**enter text; press Return/Enter**
3	**cd /tools/common/ea**	**enter text; press Return/Enter**
4	**EcCoMonitorGui /tools/common/ea *<MODE>*&** (starts the GUI to monitor the specified mode)	**enter text; press Return/Enter**
5	To update to the current status at any time, activate the **Update** button	**single-click**
6	To monitor other servers, repeat Steps 1 through 4 for other hosts their servers	
7	To exit, activate the **EXIT** button	**single-click**

7.2.2.3 Using EcCsIdPingServers to Ping All Servers in a Mode

The script **EcCsIdPingServers** script works with a *Sweeper* binary to ping the servers and clients in a mode to determine their status. Table 7.2-5 presents the steps required for using EcCsIdPingServers to ping all servers in a mode. If you are already familiar with the procedure, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedure:

1   Log in to one of the host machines.

2   At the UNIX prompt, type **cd /usr/ecs/<*MODE*>/CUSTOM/utilities**, and then press the
        Return/Enter key.
   • The prompt reflects a change to directory **cd /usr/ecs/<*MODE*>/CUSTOM/utilities**,
     where **<*MODE*>** is likely to be **OPS**, **TS1**, or **TS2**.

3   Then type **EcCsIdPingServers <*MODE*>**, and then press the **Return/Enter** key.
   • The result should appear similar to the following:
     /usr/ecs/DEV03/CUSTOM/bin/CSS/Sweeper -nsh dss2 -nsp 22822
     FoSwSweeper application started...
     We made a connection with EntryId =g0icg01:17871:12451240 ---
     EcSrTransportEcInGranServer
     We made a connection with EntryId =g0ins02:22336:6737528 --- DsHrQuitIDL
     We made a connection with EntryId =g0pls02:35211:25637 --- PlOdMsgDObj
     We made a connection with EntryId =g0dis02:48315:18311 --- DsDdRequestMgrIDL
     We made a connection with EntryId = g0ins02:17862:12461267 --- InAutoIngestIF
     We made a connection with EntryId = g0dis02:49473:13375 --- DsStReqMgrIDL
     We made a connection with EntryId = g0ins02:41566:13071 --- IoAdRpc
     We made a connection with EntryId = g0ins02:18139:12460808 --- InRequestMgrIF
     We made a connection with EntryId =g0dms03:42000:13266 --- EcSrTransportDDICT
     We made a connection with EntryId = g0pls02:22359:6737528 ---
     **   DsHrNonConfIDL681ab65e-60bc-1024-8e70-08006902a6d6**
     We made a connection with EntryId = g0pls02:22346:6737528 ---
     **   DsHrConformantIDL681ab65d-60bc-1024-8e70-08006902a6d6**
     We made a connection with EntryId =g0mss21:64657:8006 --- EcAcOrderMgr
     We made a connection with EntryId =g0mss11:41449:22898 ---
     EcSrTransportDarServer
     We made a connection with EntryId = g0icg02:17724:12445092 --- EcRgRegistry
     We made a connection with EntryId =g0mss11:41278:22739 --- InDDNTransferPkt
     We made a connection with EntryId =g0psl02:35085:25466 --- Deletion
     We made a connection with EntryId =g0pls02:35168:25584 --- SubscriptionQueue
     We made a connection with EntryId =g0mss21:64700:8059 --- MsAcUsrRequestMgr
     We made a connection with EntryId =g0mss21:64690:8059 --- MsAcRegUserMgr
     We made a connection with EntryId =g0mss21:64695:8059 --- MsAcUsrProfileMgr
     We made a connection with EntryId =g0pls02:35127:25527 --- DpPrSchedulerDObj
     We made a connection with EntryId =g0ins02:22364:6738409 ---
     **   DsHrNonConfIDL681ab654-60bc-1024-8e70-08006902a6d6**
     We made a connection with EntryId =g0ins02:22353:6738409 ---
     **   DsHrConformantIDL681ab653-60bc-1024-8e70-08006902a6d6**
     We made a connection with EntryId =g0ins02:22342:6738409 --- DsHrQuitIDL

Table 7.2-5. Using EcCsIdPingServers to Ping All Servers in a Mode

Step	What to Do	Action to Take
1	Log into one of the host machines	
2	**cd /usr/ecs/<*MODE*>/CUSTOM/utilities**	**enter text; press Return/Enter**
3	**EcCsIdPingServers <*MODE*>**	**enter text; press Return/Enter**

611-EMD-001, Rev. 02

8. Problem Management Procedures

EMD Problem Management is administered through system-level and site-level control board reviews. These control boards oversee the analysis, recommendations, and actions taken to resolve system/site problems concerning hardware, software, documentation, and procedures. Operations (OPS) Deployment and the site-level organizations resolve routine maintenance issues at the system-level and site-level, respectively, using the trouble ticket system for tracking system problems. Trouble Tickets (TT) may evolve into Non Conformance Reports (NCR), as required, which may then be utilized to generate Configuration Change Requests (CCR) to effect changes to the approved baseline. To ensure controlled change, NCRs are tracked using the DDTS in the EDF and CCRs are tracked manually by Systems Engineering.

The trouble ticket process is the first vehicle used to record and report problems with the operational system. Trouble Tickets can be generated by operations, maintenance, development, and customer personnel as well as external users. The trouble ticket process uses an automated database that tracks the resolution activities associated with each trouble ticket. Documentation that is related to the problem, and is not in electronic form, or is in electronic form at the DAAC, is distributed by the local CM Administrator (CMA), and is listed as an attachment to the trouble ticket.

The CM Administrator at each site serves as trouble ticket system administrator. Trouble Tickets initially generated at a site, the resolution of which require changes to the system level baseline, are forwarded to the SMC, where they are reviewed and *translated* into NCRs. Additionally, Trouble Tickets and CCRs that are generated at the sites, which are repaired locally, and result in site-unique extensions to the system level baseline, are forwarded to the SMC for tracking across the EMD baseline. The Problem Review Board (PRB) designee is responsible for tracking TTs after they have been received from the sites, and for propagating system problem resolutions for site visibility.

CMAs also support the activities of the local Problem Review Board. This includes generating status reports, and implementing resolutions, instructions, and changes as directed by the Board. User Services Representatives monitor trouble tickets to notify users concerning problem resolution and status. Maintenance engineers at respective levels will record all activities in the trouble ticket. This information can be used to determine critical maintenance concerns related to frequency of occurrence, criticality level, and the volume of problems experienced. The maintainability analysis will guide critical changes, volume and type of support components to be utilized, and will focus further development efforts.

This section provides an overview of the Trouble Ticketing process and defines the procedures for processing and resolving trouble ticket submissions. In addition, this section provides instructions for diagnosing network problems.

8.1 The Problem Resolution Process — An Overview

8.1.1 Internal Process

Any system user may submit Trouble Tickets using the local trouble ticketing system at any site. TT submission triggers an internal review by the site's review board. Primary objectives of the internal review are to quickly identify and correct problems that fall within the site's capability to maintain, review and validate the priority of the problem, and to elevate to the system level those problems that either exceed their capability to repair, or that require a change to the system level baseline.

Problems passed from the sites to the ECS system level, are passed by transferring the trouble ticket from the local trouble ticket system to the SMC trouble ticket system. Here they are reviewed by the EMD Problem Review Board (PRB), which hosts daily teleconferences, known as the PRB Telecon.

The permanent membership of the Problem Review Board is as follows:

    a. Chair: Problem Management Lead or designee

    b. Each DAAC: one member representative

    c. Quality Assurance: one member representative

    d. ESDIS: one member representative

The roles and responsibilities of the participants in the PRB Telecon are to:

  a) Follow a nominal agenda that includes the following discussions:

    i)   New trouble tickets.

    ii)   Deferred trouble tickets.

    iii)   Aging high trouble tickets.

    iv)   Review of all Severity 1 (Sev 1) NCRs (OPS Only).

    v)   Review of new OPS NCRs. (Full Report Only)

  b) Review severity of each NCR according to the guidelines specified in the

      Operations Class NCR Management Process (MO-1-003-5)

  c) Convert any TTs that identify a system non-conformance and have the appropriate information will be forwarded into an NCR by requesting that the Problem Management Lead forward the TT to the DDTS (Distributed Defect Tracking System). NCRs document the system non-conformance and are worked by the Sustaining Engineering staff.

  d) NCRs that are deemed to not have enough information for proper adjudication will either be rejected as incomplete or they will be deferred until a future meeting as determined by the PRB chairperson. The PRB administrator will maintain a list of deferred items and place those items on a future PRB agenda after more information is gathered.

The PRB performs a preliminary review of each trouble ticket to confirm the severity assigned by the site, the completeness of information and data relevant to the problem, and whether it requires a change to the system level operational baseline.

The PRB forwards TTs to the Development Sustaining Review Board. Comprised of a lead from Development and COTS for technical investigation, resolution proposal, and NCR preparation, when required. The CM Administrator or DAAC Support Help Desk members provide administrative support to the PRB by publishing minutes of teleconference meetings, tracking TTs and action items, and by translating TTs to NCRs.

The Sustaining Engineering Review Board has the authority to direct resolutions to trouble ticket problems that do not change, or in any way affect, the EMD operational baseline and baseline documentation. An NCR is required when the Technical Investigation (TI) determines that the operational baseline must be changed in order to correct the problem identified in the trouble ticket.

The PRB is not a voting board; the membership is appointed for the purpose of providing timely, direct technical support to the Chair, who has the decision-making responsibility and authority.

The Science Development (SCDV) CCB is not a voting board. It does have the authority and responsibility to approve Class II changes to the operational baseline. (Class II changes are described in the *Configuration Management Plan for the EMD Project* (110-EMD-001).) Specific responsibilities include:

a. Review, approve and schedule; review and backlog; or reject each NCR's proposed resolution, or cost and schedule input from the Responsible Engineer (RE).

b. Approve the schedule for the deployment of configuration changes in the form of a 'drop' to the SMC.

c. Approve the content of each block.

d. Manage and adjust the schedule and contents of each block in accordance with program priorities and the progress of NCR work-off.

e. Review the status of all backlogged NCRs on a periodic basis. Schedule NCRs for a future block as appropriate

f. Collect and report on NCR statistics.

The permanent membership of the SCDV CCB is as follows:

a. Chair: EMD Systems Engineering and Integration Team (SEIT) designee.

b. Members of the EMD SEIT and designees.

c. Each DAAC: one representative.

8.2 Problem Management Procedures

The trouble ticket system is comprised of the Remedy Action Request System, a Commercial Off-The-Shelf (COTS) product that provides a distributed trouble ticketing service which provides a common environment and means of classifying, tracking, and reporting problem occurrence and resolution to both system users and operations personnel. The trouble ticketing service:

- Provides a GUI for operations personnel to access all Trouble Ticket services.

- Provides a common Trouble Ticket entry format.

- Stores Trouble Tickets.

- Retrieves Trouble Tickets via ad hoc queries.

- Allows operations personnel to forward problems from one DAAC to another.

- Produces stock and common reports.

- Provides an interface to user's and operator's e-mail to provide automatic notification.

- Offers an application programming interface through which applications can submit trouble tickets.

- Provides summary information to the SMC from each DAAC to allow trend reports regarding trouble tickets.

- Defines a consistent "life cycle" for trouble tickets.

- Allows each DAAC a degree of customization through definition of further re-prioritization and action rules.

In addition to the functionality provided by Remedy's Action Request System, the trouble ticketing service utilizes a set of custom HTML pages ("screens") to provide registered users with the ability to submit new trouble tickets and query the current status of any of their previous entries. Access to the Trouble Ticketing System through this technique provides users an easy method for reporting problems in an environment with which most are already familiar. Additionally, as another means of trouble ticket entry, the trouble ticket system provides a text e-mail template through which automated entry of trouble tickets is possible. Support staff members are able to enter Trouble Tickets through the trouble ticket system interface for problems received via other methods (for example, phone calls).

The Remedy Action Request System also functions as the User Contact Log. Remedy's Action Request System is configured to have a separate form that contains the entries that User Services personnel enter for each contact that they receive from a user. The User Contact Log allows a trouble ticket to be initiated from a log entry: with the push of a button — the Trouble Ticket will be populated with information from the contact log.

Users submit trouble tickets to the User Services Desk. External users submit trouble tickets through the Internet [using a series of hypertext mark-up language (HTML) screens]. Site personnel submit trouble tickets via the trouble ticket system (Remedy).

User Services personnel process trouble tickets for problem resolution in accordance with local policy. Trouble tickets are first evaluated to determine the severity of the problem and assignment of on-site responsibility. Every trouble ticket is logged into the database for record keeping purposes. Trouble tickets that can be resolved locally are assigned and tracked at the local center. The Operations Supervisor reviews each trouble ticket (for priority verification and problem description) and assigns it to an appropriate Maintenance Engineer for resolution.

Matters that require external or higher level assistance and problems, the repair of which require changes to the system baseline; are escalated via the PRB Telecon for discussion and disposition. The telecon is held to coordinate trouble ticket activities within the EMD organization as well as with development, customer, and user organizations.

8.3 Using the Trouble Ticket System

1. User or Operator discovers a problem with the system (hardware, software, documentation, procedure) and documents this problem for later resolution. The submitter forwards a trouble ticket to User Services by: calling up the trouble ticket system via the Internet; phoning User Services; or sending an e-mail message to the trouble ticket system.

2. The trouble ticket is logged into the system. It automatically assigns "New" status to the trouble ticket and notifies the Operations Supervisor for assignment and prioritization. The status of each trouble ticket is the responsibility of Operations as it progresses through the resolution process.

3. Categorizing Problem Severity. The DAAC Help Desk will review the information provided by the DAAC to determine if the problem has been described in enough detail to warrant the recommended severity and category. If there is insufficient information, the DAAC submitter or point of contact will be contacted. The Trouble Ticket will be deferred to the Sustaining Engineering Review Board for further review.

   In determining severity of the problem, the DAAC Help Desk must consider the following factors:

   - Impact on the ability to ingest, process or distribute satellite data

   - Frequency of occurrence

   - Availability of an adequate work-around

   Refer to Table 8.3-1 to determine problem severity.

Table 8.3-1. Trouble Ticket Priority/NCR Severity

As Documented in NASA 420-05-03	As Used/Interpreted by the EMD Project
Category 1: System/Service cannot perform critical function or imposes major safety hazard. (Priority 1) Presents an immediate impact to development, operations, services, or data processing functions; imposes major safety hazard to personnel, systems, or space mission resources; or results in loss of one or more essential mission objectives.	**HIGH (Severity 1)**: An NCR which causes: _ Inability to perform a mission-critical function (i.e., Ingest/Pre-Processing/Archiving of Science Data, Planned Processing, Browse/Order/Distribute); _ Performance of a mission-critical function to be so degraded that production minimum goals cannot be achieved; _ A mission-critical function to be performed improperly, resulting in permanent loss of data; **and** for which no workaround exists **or** for which no workaround can be accommodated by DAAC operators given a detailed workaround procedure is documented but the procedure is inadequate based upon the complexity of the procedure, the abilities of an adequately trained and experienced operator, or both.
Category 2: System/Service substantially impaired. (Priority 2) Substantially impacts development, operations, services, or data processing functions; fails to operate within critical performance specifications; or cannot effectively or efficiently fulfill baseline requirements.	**MEDIUM (Severity 2)**: An NCR with the consequence that: _ The performance of a mission-critical function is degraded and may prevent achieving production minimum goals; _ A mission-critical function can be only partially performed, or performs improperly, resulting in temporary loss of data or incorrect data results; _ A situation (actually or potentially) severly compromises ECS mission readiness or operational integrity; _ A condition exists to produce a severely degraded mission-critical function, but a workaround will allow operations to continue temporarily without permanent loss of data or severely impaired performance/workload/schedules.
Category 3: System/Service slightly impaired. (Priority 3) Causes minor or no substantial impact to development, operations, services, or data processing functions. Support may be degraded, but mission can still be accomplished.	**Severity 3**: An NCR with the consequence that: _ A non-critical mission function (e.g., Advertising) cannot be performed, or yields incorrect results; _ Unexpected events occur which can be corrected using normal operational procedures with minimal impacts to performance/workloads/schedules _ A condition exists to produce a degraded mission-critical function, but a workaround will allow operations to continue indefinitely without severely impaired performance/workload/schedules.
	Severity 4: Improvement (Nuisance; e.g., a typo).
	Severity 5: Enhancement (Identified for next release).

4. All affected Operations Supervisors at the sites (SMC, DAACs, EOC, EDF) are notified by e-mail of the problem and solicited for inputs to problem assessment (impact) and resolution.

5. The Trouble Ticket database is updated by the CM Administrator or the DAAC Support Help Desk Staff.   The trouble ticket may be modified to reflect any new information/coordination activity.

6. The Operations Supervisor assigns the problem to a Problem Investigator for further follow-up.

7. The Problem Investigator coordinates input from developers, vendors, and external organizations to effect the local resolution. The Problem Investigator presents significant issues at the PRB Telecon.

8. The Problem Investigator updates the trouble ticket database.

9. The Problem Investigator forwards any information regarding proposed/implemented fixes to the established notification list.

10. In those cases where the problem resolution will result in a change to the operational baseline, or where the local site wishes to elevate the TT to the PRB for advice or for resolution, the site CMA forwards the TT to the SMC.

11. The proposed resolution is then presented to the Problem Review Board for review, ratification, or revision.

12. Changes that do not affect controlled configuration items may be approved and implemented by the Problem Review Board and closed.

13. All system level changes are proposed in Non Conformance Reports (NCRs). The SMC opens an NCR in the Change Request Manager and closes the trouble ticket in Remedy. Emergency fixes (High TTs) can be made locally with the approval of the local CCB and then be forwarded to the SMC in a trouble ticket.

14. The off-site problem resolution process is monitored by the EMD Problem Review Board, which may also revise the proposed solution because of any system-level effect(s).

15. The NCR may be escalated to higher levels for system and/or external elements that may be involved in the resolution process.

Table 8.3-2 provides the activity checklist for the Trouble Ticket System.

Table 8.3-2. Trouble Ticket System - Activity Checklist

Order	Role	Task	Section
1	System Users	Access the Trouble Ticket System	8.3.1, 8.4.1
2	System Users	Submit Trouble Ticket	8.3.2
3	Maintenance Engineer	Modify Open Trouble Ticket	8.3.3
4	Operations Supervisor, Maintenance Engineer	Forward Trouble Ticket	8.3.4
5	Database Administrator	Add Users to Trouble Ticket System	8.3.5
6	CMA	Modify Trouble Ticket System User Privileges	8.3.6
7	CMA	Modify Trouble Ticket System Configuration	8.3.7
8	Maintenance Engineer, CMA	Generate Reports	8.3.8
9	Maintenance Engineer	Maintain Escalation Time Table	8.3.9

8.3.1 Accessing the Trouble Ticket System

The trouble ticket system may be accessed through either HTML or Remedy. The Trouble Ticket HTML is used by both User Services and the end user to submit trouble tickets without going through Remedy. It is accessed through the web. Through HTML, the user can submit,

611-EMD-001, Rev. 02

obtain a list, and view details of trouble tickets. Complete and detailed instructions for Remedy may be found in 609-EMD-001, *Release 7.11 Operations Tools Manual for the EMD Project* and in Remedy Action Request System documentation at the company's web site:

http://www.remedy.com

The Remedy customer support web site requires user login. Contact the DAAC COTS software representative for assistance.

Through Remedy, the User clicks on the User Tool icon, which opens the form to submit, query, or work a Trouble Ticket. The Main Remedy Trouble Ticket screen is used to select the appropriate forms for submitting, modifying, or displaying a trouble ticket. The Main Page data fields are identified in Table 8.3-3.

The Remedy Action Request System provides a distributed trouble ticketing service that furnishes DAACs a common environment and the means of classifying, tracking, and reporting problem occurrences and resolutions to both system users and operations personnel.

Several timesaving features are available through Remedy: the Admin Tool, Remedy Import tool, the Hardware Information form, and the GUI Notification tool. Brief descriptions are provided in Sections 8.3.1.1 through 8.3.1.4.

Table 8.3-3. Trouble Ticket Field Description (1 of 3)

Field Name	Data Type	Size	Entry	Description
Ticket-Id	Character	15	System generated	Ticket number that is set and maintained by the system.
Ticket Status	Selection	*	Required	Status of the trouble ticket.
Assigned-Priority	Selection	*	Required	Priority of trouble ticket assigned at the site (HIGH, MEDIUM, LOW).
Short Description	Character	128	Required	Short description of the problem.
Submitter Impact	Selection	*	Required	Impact of the problem to the submitter.
Long-Description	Character	4060	Optional	Long description of the problem. Including Category
Resolution Log (End User Sees)	Diary	Unlim	Optional	General steps in the resolution of the problem.
Detailed Resolution Log	Diary	Unlim	Optional	Detailed steps in problem resolution.
Submitter ID	Character	30	Required	User Id of the Submitter.
Submitter Name	Character	30	Optional	Full Name of the Submitter.
Submitter Phone	Character	30	Optional	Phone number of the Submitter.
Submitter eMail	Character	64	Optional	E-mail address of the Submitter.
Submitter Home DAAC	Character	60	Optional	Home DAAC of the Submitter.

Table 8.3-3. Trouble Ticket Field Description (2 of 3)

Field Name	Data Type	Size	Entry	Description
History	Diary	Unlim	Optional	Upon submission or modification, the person assigned to the ticket and the ticket status will be indicated in the History field. Due to a limitation in Remedy, this information is only written when the Assigned-to and Status fields are modified.
CI	Character	30	Optional	Name of the configuration item to which the problem is associated.
Assigned-To	Character	30	Optional	Person to whom Trouble Ticket has been assigned.
Last-modified-by	Character	30	System generated	Person who last modified the Trouble Ticket.
Create-date	Date/Time	17	System generated	Date and time Trouble Ticket was created at the present site (mm/dd/yy hh:mm:ss).
Last-Modified-date	Date/Time	17	System generated	Date and time the trouble ticket was last modified (mm/dd/yy hh:mm:ss).
Related CCR	Character	60	Optional	ID of a related CCR. [If more than one CCR, separate CCRs by a space for readability.]
Key Words	Character	255	Optional	Key words to help identify this Trouble Ticket.
Problem Type	Character	30	Optional	Type of problem addressed by this Trouble Ticket (e.g., configuration error, hardware problem, software problem).
Closing Code	Character	60	Optional	Code assigned to the type of problem that necessitated the writing of this Trouble Ticket.
Closed-by	Character	60	Optional	Person who closed this Trouble Ticket.
Close-date	Date/Time	17	Optional	Date and time this Trouble Ticket was closed (mm/dd/yy hh:mm:ss).
Software Resource	Character	60	Optional	Software resource that the problem came from.
Hardware Resource	Character	60	Optional	Hardware resource that this problem came from.
Duplicate Master Id	Character	25	Optional	The Ticket-ID of the primary Trouble Ticket for the problem reported in this Trouble Ticket and its associated duplicate Trouble Tickets (other tickets reporting the same problem).
Forward-to	Character	60	Optional	Site that this Trouble Ticket was last forwarded to.
Forwarded-from	Character	60	Optional	Site that forwarded this Trouble Ticket.
Forwarded-by	Character	60	Optional	Contact person at the forwarding site.
Forward-date	Date/Time	17	Optional	Date and time Trouble Ticket was forwarded (mm/dd/yy hh:mm:ss).

Table 8.3-3. Trouble Ticket Field Description (3 of 3)

Field Name	Data Type	Size	Entry	Description
Unique-Identifier	Character	20	Optional	Unique identifier that is established at the origination site. This identifier should NEVER be changed once set.
Forwarded-to-1	Character	60	Optional	First site to have been forwarded this Trouble Ticket.
Forwarded-to-2	Character	60	Optional	Second site to have been forwarded this Trouble Ticket.
Forwarded-to-3	Character	60	Optional	Third site to have been forwarded this Trouble Ticket.
Forwarded-to-4	Character	60	Optional	Fourth site to have been forwarded this Trouble Ticket.
Associated Contact Log Id	Character	30	Optional	ID number of the associated Contact Log entry for this Trouble Ticket.

*Note, the size of a field with a "selection" data type can vary and the size is automatically adjusted to the size of the item selected from the selection list.

In addition to the fields described in the preceding table, the **RelB-Trouble Tickets** form provides the following active links:

- **Forward**—Forwards the trouble ticket to the site specified in the **Forward-to** field (**Ticket Status** field must be set at **Forwarded**).

- **Hardware Information**—Opens a window that is associated with the trouble ticket to hold hardware information.

- **List All Masters**—All trouble tickets that are duplicates of each other have one master. This button lists all master trouble tickets.

- **List This TT's Duplicate(s)** -- List all trouble tickets that are duplicates associated with this trouble ticket.

- **Go to Contact Log** -- If the trouble ticket was created from a contact log (User Services), then this button opens a window to that contact log.

8.3.1.1 Remedy's GUI Admin Tool

Most Remedy administrative functions are accomplished using the Remedy Administration tool, implemented in the Windows environment on a personal computer. The log-in dialog provides an **Accounts . . .** button for selecting the user account and servers to be addressed in the administration session. When the account and servers have been selected and the log in is complete, the Remedy Administrator Server Window is displayed. In this window, the server directory structure may be expanded and an object type may be selected for performing administrative functions on various "objects" such as forms and groups (see Sections 8.3.5, 8.3.6, and 8.3.7).

8.3.1.2 Remedy Import Tool

The Remedy Import tool is used to import existing entries rather than retyping information manually. It also enables the user to import entries into a form from a file generated by the Admin tool. More information on the Import tool is available in Remedy Action Request System documentation at the company's web site, as described in Section 8.3.1.

8.3.1.3 Remedy's Hardware Information Form

Detailed hardware information can be provided beyond what can be entered on the Trouble Tickets form by using the Hardware Information form. The User Tools Hardware Information form provides the vehicle to add a description of a hardware problem that corresponds to a trouble ticket. Through this form, the user can enter detailed information about failed hardware components (e.g., part and serial numbers) and the actions taken to correct the problem. This form is accessed by opening RelB-Hardware Information form or via a Hardware Information link from the Trouble Tickets form.

8.3.1.4 Remedy's GUI Notification Tool

The GUI Notification Tool is used as an alternative to email notification to notify the user of a Remedy event. It allows properties and options to be modified via pull-down menus. Examples of GUI notification include a beep, a pop-up window, and a flashing message. In addition, both an email and a GUI notification can be sent if the site so desires.

8.3.2 Submitting a Trouble Ticket

When a problem is either found by or reported to User Services, follow the procedure applicable to your system, to create and log trouble tickets. Trouble tickets can be submitted via HTML or via Remedy's user tool – RelB-Trouble Tickets form. Remedy's Contact Log form is used to classify, track, and report contacts of system users and operators and also to submit a trouble ticket from a log entry. E-mail is another method of submitting a trouble ticket. The template is available from your System Administrator.

1.   For HTML submission:

   a)   Access HTML Trouble Ticketing Main page.

   b)   Select Submit link, which opens the Submit page.

   c)   Fill out the impact, short description, and detailed description fields.

   d)   Select Submit.

2.   For submission through Remedy:

   a)   Access Remedy User Tool by logging in to the Remedy host and executing the command **aruser &**.

   b)   Access the RelB-Trouble Tickets form by selecting **Open** from the **File** menu, highlighting **RelB_TroubleTickets**, and clicking the **New** button.

c) Fill in those fields as specified in Table 8.3-3, Trouble Ticket Field Description.

d) Select **Save** from the **Actions** menu (or click on the **Save** button).

3. For submission from a Remedy Contact Log entry:

a) Open RelB-Contact Log form.

b) Fill out Contact Log ID and Contact Information. If the contact is a registered Remedy user, the contact information is filled out automatically.

c) Fill in Short Description (limit is 128 characters).

d) Click on **Create TT** button.

4. For submission via E-mail:

a) Obtain Template from your System Administrator.

b) Address the message to arsystem@_____._____._____.

c) Copy template into message area. DO NOT INCLUDE AS AN ATTACHMENT. DO NOT ALTER TEMPLATE. The template is presented in Figure 8.3.2-1. The # sign indicates comments, which are not read by Remedy. **Enter data as indicated in Figure 8.3.2-1.** Send message.

```
#
#   File exported Wed Feb 28 19:01:27 1996
#
Schema: RelB-Trouble Tickets
Server: remedy server name
Login:                                        Field ID internal
Password:                                     to Remedy

        Short Description !        8!:             Default
         Submitter Impact !536870922!: Low        value
# Values: Low, Medium, High ————————————— Select one
        Long-Description !        9!:
            Submitter ID !        2!:
          Submitter Name !536870917!:        Enter data after
         Submitter Phone !536870918!:        colon
        Submitter e-mail !536870921!:
     Submitter Home DAAC !536870919!:
```

Figure 8.3.2-1. Trouble Ticket E-mail Template

8.3.3 Reviewing and Modifying Open Trouble Tickets

Trouble tickets may need to be modified based on better understanding of the nature of problems defined and revised resolutions from the Maintenance Engineer investigations, Sustaining Engineering inputs, Developer inputs, Problem Review Board decisions, Change Control Board

decisions, and/or Failure Review Board decisions. The results will be factored into revisions and/ or additions to the Trouble Ticket log.

1.   For HTML Review and Modification of Trouble Tickets:

   a)   Access HTML Trouble Ticketing Main (see Section 8.4). Trouble Tickets can be *submitted, queried or modified*.

   b)   Select List link, which opens the List page and shows each Trouble Ticket's Identification, Short Description, and Status.

   c)   Select the Trouble Ticket Id to get a more detailed description of that particular Trouble Ticket.

2.   For Reviewing and Modifying Trouble Tickets through Remedy:

   a)   Access Remedy User Tool (see Section 8.3.2.2a).

   b)   Access the RelB-Trouble Tickets form by selecting **Open** from the **File** menu, highlighting **RelB_TroubleTickets**, and clicking the **Search** button.

   c)   In the resulting **Modify RelB-TroubleTickets** window, enter a search criterion and then select **Search** from the **Actions** menu (or execute that menu selection with no search criterion entered, to obtain a list of all Trouble Tickets, and then select one to modify).

   d)   Review and/or modify the Trouble Ticket fields as necessary.

   e)   Select **Save** from the **Actions** menu (or click on the **Save** button).

8.3.4 Forwarding Trouble Tickets

Trouble ticket administrative reports are forwarded for local and system-wide usage. The trouble ticket contains all forwarding information; once forwarded, it goes to the RelB-TT-ForwardToSite holding area (transparent to the user). The RelB-TT-Sites schema is used to indicate the site name and email address to be used in forwarding. To forward a trouble ticket:

1. Access the Trouble Ticket to be forwarded (see 8.3.3a to 8.3.3c).

2. Select **Forwarded** from the pull-down menu in the **Ticket Status** field.

3. Select the TT's destination using the **Forward-to** field pick list.

4. Select the **Forward** button.

5. Select **Save** from the **Actions** menu (or click on the **Save** button).

8.3.5 Adding Users to Remedy

The TT Administrator uses the Remedy User form to grant access to the Remedy tool. Users who leave the EMD program can be deleted. Further information concerning access control and license elements is available in Remedy Action Request System documentation at the company's web site, as described in Section 8.3.1.

There are three classes of licenses that can be assigned to users:

- The **Read** class allows users to submit requests (e.g., Trouble Tickets, User Contact Log records) and modify their own requests. Thus, there are no license restrictions on the number of users who can be granted permission to create and read trouble tickets.

- The **Fixed Write** class provides the capabilities of the Read class plus the ability to modify existing requests submitted by others. A Fixed Write license assigned to a user is always reserved for that user. The AR system administrator must have a Fixed Write license.

- The **Floating Write** class provides the capabilities of the Fixed Write class, but these licenses are not reserved to a single user; instead, they are available on a first-come, first-served basis. Each DAAC has five Floating Write licenses. It is appropriate to assign Floating Write licenses to those who will need to modify Trouble Tickets (e.g., those who will be assigned to work and resolve problems). The limit of five licenses should pose no problem, as it is unlikely that there will be many instances in which more than five people will be trying to modify Trouble Tickets at one time.

Information on defining access control, including using the **User** form (a "form" represents a table in the Remedy database) to add registered users, is available in Remedy Action Request System documentation at the company's web site, as described in Section 8.3.1.

8.3.6 Changing Privileges in Remedy

Changing privileges in Remedy, or controlling privileges of those who have access to Remedy, is done by the CM Administrator. There are numerous Remedy privilege groups for the system, and a change to the privileges of any group requires an approved Configuration Change Request (CCR). Access privileges determine which forms a user may open and specify whether a user may change information in a field or merely view it.

Information on adding groups (i.e., using the **Group** form to create access control groups to which users may then be assigned) is available in Remedy Action Request System documentation at the company's web site, as described in Section 8.3.1.

A user's privileges may be changed in two ways:

- Changing the group to which the user is assigned.

- Changing the access privileges of the group.

The **User** form is used to implement group assignment. To change the access privileges of a group, you use the **Admin** tool. There are two primary ways to change the access privileges of a group. Both ways use the **Group Permissions** window. This window is obtained by selecting **Groups** on the **Remedy Administrator – Server Window** and then double clicking on one of the listed groups. One way to change access privileges is to use the **Form Permissions** tab of the **Group Permissions** window to select which forms are available to members of the group when they access the **Open** dialog box of the **User** form. The second way is to use the **Field**

Permissions tab to specify which fields of a form the members of the group may change and which fields they may view but not change. Groups have already been created to accommodate all privileges needed by Remedy Users for Release B. These groups are identified in Table 8.3-4.

Table 8.3-4. Table of Access Control Groupings

Groups	Description	Access Type
Operator	Submits trouble ticket internally.	Change
User Services	Submits trouble ticket internally for user.	Change
Operations Supervisor	Assigns problem priority and resolution responsibility. Can forward trouble ticket to another site.	Change
Resource Manager	Assigns problem priority and resolution responsibility. Can forward trouble ticket to another site.	Change
Resolution Technician	Attempts to resolve problem.	Change
Problem Review Board Chair Person	Reviews proposed solutions.	Change
Administrator	Adds groups and users. Changes permissions. Sets escalation times. Sets menu items. Etc.	Change
Sub-Administrator	Same functions as Administrator but only with certain Schemas.	Change
Browser	Read only permission.	Read
Customize	Can use all features of the customize facility.	Change
Submitter	Place holder for anyone that submits a trouble ticket.	NA
Assignee	Place holder for anyone that is assigned a trouble ticket.	NA
Public	Read only permission. Guest users are automatically put in this group.	Read
NotifyNewEscal	Everyone that will be notified on an escalation due to trouble ticket being in "New" status.	Read
NotifyAssignedEscal	Everyone that will be notified on an escalation due to trouble ticket being in "Assigned" status.	Read
NotifySolPropEscal	Everyone that will be notified on an escalation due to TT being in "Solution Proposed" status.	Read
NotifyImpSolEscal	Everyone that will be notified on an escalation due to trouble ticket being in "Implement Solution" status.	Read
NotifySolImpEscal	Everyone that will be notified on an escalation due to TT being in "Solution Implemented" status.	Read

8.3.7 Modifying Remedy's Configuration

RelB-Trouble Ticket forms' pulldown menus can be customized. Customization is achieved through the Admin Tool by modifying the RelB-Menu-Closing Codes, RelB-Menu-Hardware Resources, RelB-Menu-Software Resources, RelB-Menu-Key Words, RelB-Menu-Problem Type, or Sites forms.

1. To modify the Remedy environmental variables, refer to the Remedy Action Request System documentation at the company's web site (as described in Section 8.3.1).

    NOTE: No administrative configuration should be made without proper configuration change approval.

8.3.8 Generating Trouble Ticket Reports

A set of predefined reports is available through Remedy. These reports are trouble ticket administrative reports generated for local and system-wide usage. There are several types of predefined reports, including:

- Assigned-to Report – provides a report of the number of Tickets assigned to technicians.

- Average Time to Close TTs – provides a report of the average time to close trouble tickets.

- Hardware Resource Report – provides a report sorted and grouped by Hardware Resources and Closing Codes.

- Number of Tickets by Status – provides the number of Trouble Tickets grouped by Status.

- Number of Tickets by Priority – provides the number of Trouble Tickets grouped by assigned priority.

- Review Board Report – provides a report of the details of TTs for the TT Review Board.

- SMC TT Report – provides a report to be sent to the SMC.

- Software Resource Report – provides a report sorted by Software Resources and their Closing Codes.

- Submitter Report – indicates by submitter the number and type of trouble tickets in the system.

- Ticket Status Report – provides a report sorted and grouped by Ticket Status.

- Ticket Status by Assigned-to – provides a report sorted and grouped by the last person assigned to a Trouble Ticket.

Most of the time, you will probably select a report from the list, using the **Report** window. If you need a custom report, it is possible to select data from the database and create a report using an appropriate separate software application (e.g., a spreadsheet program).

8.4 Using Hypertext Mark-up Language (HTML) Screens

The hypertext mark-up language (HTML) Trouble Ticket Main Screen ("ECS Trouble Ticketing: Menu") provides an introduction on how to use the Trouble Ticketing HTML, and is used by registered users to go to either the Submit page or List page.

Selecting **Submit a Trouble Ticket** will bring up the Trouble Ticketing Submit screen.

Selecting **List the [username] Trouble Tickets** will bring up the Trouble Ticketing List screen.

Help on the Trouble Ticket HTML screens is available by clicking on the Trouble Ticket Help icon at the bottom of the screen.

8.4.1 ECS Trouble Ticketing HTML Submit Screen

The HTML Trouble Ticket Submit screen is used by registered users to submit a Trouble Ticket.

Table 8.4-1 below provides a description of the Trouble Ticket HTML Submit Screen fields.

Table 8.4-1. Trouble Ticket HTML Submit Screen Field Description

Field Name	Data Type	Size	Entry	Description
ID	character	30	System generated	Submitter Id
Name	character	30	System generated	Submitter Name
E-mail address	character	64	System generated	Submitter E-mail Address
Phone	character	30	System generated	Submitter Phone Number
Home DAAC	character	60	System generated	Submitter Home DAAC
Impact	selection	4	Required	Impact to Submitter
Short description	character	125	Required	Short description of problem
Detailed problem description	character	245	Optional	Long description of problem

When the information is completed, the user can submit the Trouble Ticket by clicking on the **Submit** button on the lower half of the screen. The Problem Information Fields can be cleared by clicking on the **Reset** button. The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.2 ECS Trouble Ticketing HTML Success Screen

The HTML Trouble Ticket Success screen is used by registered users to ensure successful submission and report Trouble Ticket Id.

From this screen, the user is provided with the following information/options:

- Confirmation that the trouble ticket was successfully submitted, the trouble ticket identification number, and who submitted the trouble ticket.

- Notification that an E-mail message has been sent to the user indicating that a Trouble Ticket has been submitted and when it was closed. Selecting **this Trouble Ticket** will open the Trouble Ticket Detailed Screen.

- Instructions telling the user how to check the progress of Trouble Ticket resolution.

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.3 ECS Trouble Ticketing HTML List Screen

The HTML Trouble Ticket List screen is used by registered users to List Trouble Tickets for a user and links the listed Trouble Ticket Number to the Trouble Ticket Detailed Screen.

Table 8.4-2 below provides a description of the Trouble Ticket HTML List Screen fields.

Table 8.4-2. Trouble Ticket HTML List Screen Field Description

Field Name	Data Type	Size	Entry	Description
Trouble Ticket Number	character	15	System generated	Trouble Ticket Id
Problem Short Description	character	125	System generated	Short Description of Problem
Status	character	20	System generated	Status of Trouble Ticket

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.4 ECS Trouble Ticketing HTML Detailed Screen

The HTML Trouble Ticket Detailed screen is used by registered users to see a more detailed output of a Trouble Ticket.

Table 8.4-3 below provides a description of the Trouble Ticket HTML Detailed Screen fields.

Table 8.4-3. Trouble Ticket HTML Detailed Screen Field Description

Field Name	Data Type	Size	Entry	Description
ID	character	30	System generated	Submitter Id
Name	character	30	System generated	Submitter Name
E-mail address	character	64	System generated	Submitter E-mail Address
Phone	character	30	System generated	Submitter Phone Number
Home DAAC	character	60	System generated	Submitter Home DAAC
Status	selection	4	System generated	Status of Trouble Ticket
Impact	selection	4	System generated	Impact to Submitter (low, medium, high)
Short description	character	125	System generated	Short description of problem
Detailed problem Description	character	245	System generated	Long description of problem
Log	character	unlim.	System generated	Diary of problem resolution

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.5 ECS Trouble Ticketing HTML Help Screen

The HTML Trouble Ticket Help screen is used by registered users to get help with the HTML screens.

This screen provides general information on the following:

- Index -- links that scroll the screen to the Introduction, Submit Page, and List Page sections listed below.

- Introduction – provides information about the Trouble Ticket Help page.

- Menu Page – describes the Trouble Ticketing Menu page.

- Submit Page – describes the Trouble Ticket Submit page.

- Success Page – describes the Trouble Ticket Success page.

- List Page – describes the Trouble Ticket List page.

- Detailed Page - describes the Trouble Ticket Detailed page.

8.5 Emergency Fixes

Emergencies may be in real time with the understanding that the trouble ticket system must be brought up-to-date as soon as possible after implementing the repair. The example presented below, involves a hardware failure. The problem needs to be resolved quickly to bring a system back into operation. The resolution requires emergency replacement of a component that is of a later version than is contained in the original equipment. The scenario is summarized in Table 8.5-1.

Scenario —Example of an Emergency Change Procedure

It is 7:00 on a Saturday evening. The DAAC operator detects a problem with the automated tape library (ATL) and reports the problem to the trouble ticket system. The trouble ticket is routed to the System Administrator, who confirms that the system will not operate and notifies the site Maintenance Engineer. After running further diagnostics, the Maintenance Engineer reports the problem and symptoms to the OEM's maintenance desk. The original equipment manufacturer (OEM) maintenance representative arrives and concludes that a controller card has failed. The only card the OEM has immediately available is of a later version and no spares are available on site. It will be Monday at the earliest before a replacement board of the same revision level can be located. The site maintenance engineer reports this to the operations Crew Chief (i.e., shift leader) for a decision.

The DAAC cannot afford to have the ATL down until Monday. The Crew Chief calls the DAAC manager at home, apprises him of the situation, and obtains approval to replace the board with the later version if tests conclude that it works properly. The OEM's maintenance representative installs the board. The site's sustaining engineer tests the new controller board, finds that it works properly, and brings the ATL back on-line. The sustaining engineer updates the trouble ticket to document the configuration change and the authority for the change, and forwards it to the site CMA. The site Maintenance Engineer updates the property record with the model, version, and serial number of the new board.

The site CM Administrator reviews the trouble ticket, and presents it to the local CCB for approval. The CMA then updates the Baseline Manager with the new configuration and TT number authorizing the change. At this point, the site is operational at variance from the system baseline (i.e., site unique) and is at risk of losing maintenance support from OPS Deployment.

The site CMA forwards the trouble ticket to the SMC, presented to the PRB for priority review and is solved or translated into an NCR. The PRB reviews all emergency TTs to assess whether there may be impacts to the system and/or applicability to other sites. The SCDV CCB monitors all open NCR promotion and approves them for closure.

In the event that it is later discovered that the new version controller board has adverse impacts when operating in the system configuration, a board of the original version will have to be obtained to replace the newer version. In such cases, the action will be recorded on a new trouble ticket, citing the previous CCR.

Table 8.5-1 summarizes emergency procedures that might be taken during an after-hours, over-the-weekend emergency hardware failure.

Table 8.5-1. Example of Emergency Change Procedure

Operator/User	System
Operator prepares trouble ticket to report ATL controller failure.	Trouble ticket recorded.
System Administrator and Maintenance Engineer confirm ATL controller failure, call ATL maintenance vendor, report call and time in Trouble ticket.	Diagnosis and vendor call recorded in trouble ticket.
Maintenance vendor isolates failure to the controller card. The later version card is the only card available.	
Crew Chief notified of situation and decision needed to bring ATL up to full operating capability. Approves use of the newer version card, records decision in the trouble ticket, forwards trouble ticket to Sustaining Engineer.	
Maintenance vendor installs card, tests using hardware diagnostics. Crew Chief authorizes controller to be brought back on-line.	
Maintenance Engineer records card installation by model/version into the trouble ticket.	Trouble ticket action recorded.
Sustaining Engineer reads trouble ticket and prepares for discussion at 8:30 am meeting. Updates the TT.	Install action recorded in TT. TT routed to the CMA.
CMA updates site baseline, forwards TT to the CCB. When CCB approves the action, CMA forwards to SMC.	Site ATL baseline updated in Baseline Manager.
PRB reviews emergency NCR, checks for applicability to other sites.	

611-EMD-001, Rev. 02

This page intentionally left blank.

9. Configuration Management Procedures

The prepared procedures are applicable to all EMD hardware, software, and firmware components of systems or subsystems developed and acquired by the EMD contract and/or delegated to configuration management control by the operational site-level organizations. The procedures are applicable to all items maintained by the EMD Sustaining Engineering Organization in support of EMD mission-specific projects and multiple mission-specific institutional facilities. The procedures are not applicable to those entities controlled by higher level ESDIS Project Office CM Plans. CM procedures already in place may be used by the contractor subject to direction from the Change Control Board (CCB) chairperson.

Some major features of the approach being described here include:

- Customers participate in establishing the procedures;

- The SCDV CCB performs a support role for ESDIS and its designated on-site CCBs by processing system-level CCRs & Trouble Tickets;

- Prioritization, automated tools, and procedures are used for handling change requests;

- Diverse/Strategic representation at hierarchical CCBs facilitates a path for speedy escalation/resolution of problems/issues;

- Local organizations have the needed autonomy to accomplish their mission with the minimum necessary outside intervention to promote timely resolution of local problems and enable timely production of data products;

- Proper use and deployment of CM database assets to support all CCBs allows management monitoring, control, and analysis of activities;

- Coordination with the Failure Review Board allows coordinated response to problems and filtering of prioritized issues; and

- Common CM tools will be used in all elements of the EMD Project during operations.

The procedures are organized into seven major sections that address the flow-down of procedures from the EMD system-level to the site-level with references to site-tailoring of procedures where applicable. The topics include (Section 9.1) configuration identification, (Section 9.2) change control processes, (Section 9.3) configuration status accounting, (Section 9.4) configuration audits, (Sections 9.5 and 9.6) Software CM Manager (ClearCase), and (Section 9.7) Baseline Manager (ClearCase).

9.1 Configuration Identification Procedure

9.1.1 Purpose

The purpose of configuration identification during maintenance and operations is to incrementally establish and maintain the definitive basis of control and status accounting for the EMD control items. To accomplish configuration identification for both hardware and software, the configuration management (CM) administrator (CMA) shall ensure the maintenance of each EMD configuration controlled item in an operational baseline by executing the following tasks:

a. Assign identifiers to configuration items (CIs) and their component parts and associated configuration documentation, including revision and version number where appropriate. Assign serial and lot numbers, as necessary, to establish the CI effectivity of each configuration of each item of hardware and software.

b. Follow EMD developer guidelines as referenced below in Section 9.1.3.

c. Follow vendor nomenclature for COTS items.

d. Apply operation and maintenance (O&M) version name extensions to EMD modified item nomenclature following the rules in Section 9.1.4.

e. Follow author-designated version control and nomenclature for documents and follow guidelines from the EMD Librarian (cf. Chapter 20, *Library Administration*)

f. Support the EMD Librarian's efforts to maintain linkage of the EMD documentation to EMD configuration items in the Baseline Manager tool. Ensure that the marking and labeling of items and documentation with their applicable identifiers enables correlation between the item, configuration documentation, and other associated data.

g. Maintain a release system for configuration changes (cf. Section 9.2, *Configuration Change Control Procedures).*

h. Maintain views of operational baselines using the Baseline Manager tool (cf. Section 9.7).

9.1.2 Applicability

All EMD CM Administrators and support personnel.

9.1.3 References

ESDIS CM Plan

Configuration Management Plan for the EMD Project   110-EMD-001

EMD Software Build Process   CM-1-045

Data Identification Numbering   DM-002

DoD Mil-Std-973

9.2 Configuration Change Control Procedures

9.2.1 Purpose

The ESDIS CCB chartered Change Control Boards (CCBs) shall apply configuration control measures to all the EMD configuration items and the associated documentation prior to the time it is baselined for operations. The CCBs shall apply configuration control measures to accomplish the following objectives:

a. Ensure effective control of all CIs and their approved documentation.

b. Provide effective means, as applicable, for (1) proposing engineering changes to CIs, (2) requesting deviations and waivers pertaining to such items, (3) preparing notices of revision, and (4) preparing Specification Change Notices.

c. Ensure the implementation of approved changes.

9.2.2 Applicability

All ESDIS chartered EMD CCBs.

9.2.3 References

ESDIS CM Plan

EMD Configuration Management Plan   110-EMD-001

CCB Change Control Process   CM-004

9.2.4 Procedures

9.2.4.1 Configuration Change Request Preparation

The Configuration Change Request (CCR) form in Figure 9.2.4-1 has been developed as a medium for the drafting of CCRs throughout the EMD Operations Deployment environment for changes processed locally at EMD site-level chartered CCBs at the SMC, EOC, and DAACs (GSFC, LaRC, EDC, NSIDC). For changes processed by the ESDIS Change Control Board (CCB), use ESDIS own forms. For EMD CCBs, use the new EMD Electronic CCR Tool. There are numbered items on the form that correspond exactly to the data entry required to be performed by the respective originator of the CCR . Each CCB will have unique CCR identification sequence numbers. Each CCB can forward CCRs and reports from the Change Request Manager to SMC, which processes system-level CCRs for ESDIS CCB. The ESDIS CM Plan will determine the charter of the respective CCBs and thus the scope of CCR issues to be addressed by the site CCBs.

Earth Observing System Data and Information System (EOSDIS) Core System (ECS) Configuration Change Request (CCR)		

Earth Observing System Data and Information System (EOSDIS)
Core System (ECS)
Configuration Change Request (CCR)

1. **Configuration Control Board (CCB)** ESDIS:___ ECS:___ SMC:___ DAAC: GSFC___, LaRC___, EDC___, NSIDC___,				2. CCR No.	
3. Submitted Date:	4. Revision	5. Priority Emergency☐ Urgent ☐ Routine ☐	6. Change Class		7. Status

8. CCR Title:

9. Originator: Org: e-mail: phone:

10. Approval:_____   _____
signature   date

11. Reason for Change

(indicate attachment ___)

12. Description of Change

(indicate attachment ___)

13. Impact Analysis:
Cost: ☐ None   ☐ Small   ☐ Medium   ☐ Large
(Not exceeding $100,000)   ($100,000 to $500,000)   (Over $500,000)
Evaluation Engineer: Org: e-mail: phone:
Impact Evaluators: ESDIS___; ECS Dev___; SEO___; SMC___; DAACs: GSFC___,
LaRC___, EDC___, NSIDC___, EOC___;
Others_____   (indicate attachment ___)

14. Comments: (Indicate Sites/ Organizations Affected)

(indicate attachment ___)

15. Board Action: ☐Approved ☐ Withdrawn ☐Disapproved ☐Deferred Until_____
date
Further Action Required: ☐ ECP☐ Waiver ☐Deviation☐ Tech Direction ☐Contract Mod.
☐DCN Other:_____

16. CCB Approval Chair:_____ _____ signature date	17. CCR Implemented CM Admin. signature:_____ date:_____

Figure 9.2.4-1. Configuration Change Request (CCR) Form

The following enumerated text corresponds to the numbered items on the CCR form:

(1)   **Change Control Board (CCB) --** The designated CCB is checked-off for changes processed by the ESDIS Change Control Board (CCB) and its EMD site-level chartered CCBs at the SMC, EOC, and DAACs (GSFC, LaRC, EDC, and NSIDC).

(2) **CCR Number --** The unique serialized CCR number is applied at each site.

(3) **Submitted Date --** The date that the CCR was prepared is documented.

(4) **Revision --** The current revision is designated for tracking changed versions of original CCR.

(5) **Priority --** The priority level of the CCR is assigned. Emergency CCRs may have already been implemented on a temporary basis by the Trouble Ticket Review Board (TTRB) with concurrence from the CCB Chair who later receives the CCR to document /implement the permanent change. Urgent items will be reviewed by the next CCB meeting. Routine items will be reviewed as soon as the schedule permits.

(6) **Change Class --** Change Classes are either I or II. Class I will be handled by ESDIS-only because of cost, schedule, and/or mission impacts that may require requirements changes. Class II items do not affect mission requirements, but may have cost and/or schedule implications which affect maintenance, operations, procedures, documentation, site-tailored items, COTS implementation, site installations of core system changes, science SW changes, etc.

(7) **Status --** Table 9.2.4-1 is a summary of the CCR states maintained by the Distributed Defect Tracking System (DDTS) application SW database tool that implements the Change Request Manager. Note that the hard copy form will not be updated chronically, but will be kept in the master suspense file of the CM Administrator until closed-out with a stamp (item #7 & 15) and appropriate signatures (viz., items 16, & 17).

Table 9.2.4-1. State Table Composition in DDTS Format

State Code	Available States	State Assigned
S	Submit	Submitted
N	New	New
A	Assign-Eval	Assign-Eval
O	Assign-Implement	Assigned-Implement
R	Implement	Implemented
T	Assign-Verify	Assigned-Verify
V	Verify	Verified
C	Close	Closed
D	Duplicate	Duplicate
F	Forward	Forwarded
P	Defer	Deferred

Explanations of DDTS' State Table Content:

Uppercase character in the 1st column is the character stored in the change request record to indicate what state a change request is in. DDTS uses this character to go into the table and extract the descriptive name for display in reports.

Names in the second column are the state in the present tense. They are shown on the DDTS list of states that are available for selection during input. It's also used by some of DDTS' query and report code. It facilitates querying based on descriptive names opposed to a single letter.

Names in the third column are the state in the past tense. They are shown on the DDTS change request record. It's also used by some of DDTS query and report code. It facilitates querying based on descriptive names opposed to a single letter.

Definitions (a state is the stage that a proposed change has reached in its life cycle.):

New - the initial state for all newly entered change requests.

Assign-Eval- state entered when the change request is being assigned to an engineer for evaluation/analysis.

Assign-Implement- state entered when the change request is being assigned to an engineer for development.

Implement-state entered when the proposed change has been developed.

Assign-Verify-state entered when the developed change is being assigned to an engineer for verification testing.

Verify- state entered when a developed change has been tested and verified that it functions properly.

Close- state entered when all activity specified in the change request has been completed or that the approval authority has decided to close it prior to completion of all activity.

Duplicate-state entered when a change request is determined to be a duplicate of an existing change request. Duplicate change request identifies change request being duplicated.

Forward-state entered when a change request needs to be forwarded to another DDTS defined project (In DDTS terminology, a project is a grouping of change requests. For example, a change request from a site project can be forwarded to an EMD project.).

Defer- state entered when activity on a change proposal has to be postponed.

(8)   **CCR Title --** The CCR title is supplied by the originator.

(9)   **Originator --** The originator name, organization, e-mail address, and phone number is given.

(10)   **Approval --** The CCR is approved by the designated management authority that is assigned by the CCB. This sponsorship requirement acts as a primary filter to eliminate

from consideration those CCRs that cannot be implemented or which have no EMD site management support.

(11)   **Reason for Change --** The reason for the change is narrated on the form and/or the designated attachment.

(12)   **Description of Change --** The proposed implementation of the change is narrated along with any known impacts, resources, and expenses to be incurred.

(13)   **Impact Analysis --** Impact analysis is documented in the form of Figure 9.2.4-2. The impact analysis is collected by the CCB Chair appointed Evaluation Engineer in coordination with the CM Administrator who maintains the CM records and assembles the review package for the CCB. The Evaluation Engineer documents the list of Impact Evaluators and derives and/ or verifies cost, technical, and schedule impact of the proposed change based on all inputs received. The results of the coordinated CCR Impact Analysis inputs are presented in the CCR Impact Summary form shown in Figure 9.2.4-3 part of the CCR review package.

(14)   **Comments** -- Comments are added to the CCR to summarize sites and/ or organizations affected by the CCR. Additional comments may address proposed CCB dispositions and recommendations to be indicated by resolutions in item #15.

(15)   **Board Action --** CCB actions and follow-up actions that will be facilitated and tracked by the CM Administrator are indicated. Possible CCB dispositions are given as approved, withdrawn, disapproved, and deferred (pending follow-up activities by the indicated schedule date). Further actions are indicated as:

> Engineering Change Proposal (ECP)- changed scope of contract requirements
>
> Waiver-- declaration that certain contract requirements no longer apply
>
> Deviation-- change of contract terms or substitution of terms or deliverable requirements
>
> Technical Direction-- order by Contracting Officer's Technical Representative (COTR) to perform certain tasks within the scope of the contract
>
> Contract Modification-- changes to the terms of a contract
>
> Document Change Notice (DCN)- notification of changes to published documents
>
> Others-- Engineering Change Notice, Change Order, Escalate to higher CCB authority, etc.

(16)   **CCB Approval --** CCB approval signature authority by CCB Chair or designate.

(17)   **CCR Implemented --** This signature and close-out stamp (item #7) are executed by the CM Administrator witnessing the completion of the CCR implementation process which is tracked in the Change Request Manager automated tool DDTS and updated in Baseline Manager for affected version control status changes.

CCR Impact Analysis

Responder Request Number: ____
Responder:_____
Responder Point of Contact:
     address:_____
     _____
     phone:_____
     e-mail:_____
CCB Schedule Date:_____
CCR Number:_____
CCR Log Date:_____
CCR Originator:_____
CCR Originator Point of Contact:
     address:_____
     _____
     phone:_____
     e-mail:_____

Evaluation Engineer:_____
Evaluation Engineer Point of Contact:
     address:_____
     _____
     phone:_____
     e-mail:_____
Requested Return Date:_____

Rough Order of Magnitude (ROM) Impact Analysis

Basis of Estimate:

Technical Assumptions and Comments:

Cost Impact:
     None [ ]
     Small [ ] < $100,000
     Medium [ ] $100,000 < x < $500,000
     Large [ ] > $500,000

Schedule Impact:

Technical Assessment: (Your impact analysis should consider the implementation approach; interfaces affected; HW or SW changes required; documentation changes required-- change from/to pages; suggested alternatives, if any; and impact to security features. If your system is not impacted, please provide that information to the CM Administrator.)

Comments:

Signed:_____
                    (Responder)

Date:_____

Figure 9.2.4-2. EMD CCR Impact Analysis

CCR Impact Summary

Evaluation Engineer:_____
Evaluation Engineer Point of Contact:
          address:_____
          _____
          phone:_____
          e-mail:_____
CCR Board Date:_____

Resources Summarized:

Technical Summary:

ROM Summary (BOE, Cost, and Schedule):

Recommendation:

Signed:_____
          (Evaluator)

Date:_____

Figure 9.2.4-3. EMD CCR Impact Summary

9.2.4.2 Change Control Board Process (System and Site-level CCBs)

The EMD organization will provide administrative and technical support services for the CCB at each site. Each site's CCB is controlled by the host site organization and provides the authority and direction for the EMD contractor to modify the operational baseline. The ESDIS CCB has chartered an EMD Review Board to coordinate EMD system-level changes and problem management via the EMD contractor and on-site Review Boards that also act as site CCBs. This is illustrated using the CM Administrator's workflow for the sustaining engineering support of the EMD Review Board in Figure 9.2.4-4 and the On-Site CM Administrator's workflow for sustaining engineering support of the on-site CCB in Figure 9.2.4-5. The problem management process was discussed in detail in Chapter 8 of this document. Both diagrams illustrate the flow of CCRs through the respective CCBs with inputs from the review boards and evaluators that determine the disposition of proposed changes. Details of this process are given below:

System-level Change Control Procedures

(The enumeration corresponds to the diagram of Figure 9.2.4-4)

(1)   Configuration Change Requests are received by the EMD CM Administrator from all sources with regard to the operational EMD Core System as described in Section 9.2.4.1. These changes designated as from other sources could involve system enhancements, procedures, interfaces (both external and internal), documentation changes, etc. that are not the subject of contemporaneous problem reports which would be first deliberated by the Trouble Ticket Review Board (TTRB) and/or Failure Review Board as explained below.

(2)   Proposed common baseline changes will be proposed based on Trouble Ticket (TT) resolutions obtained from the respective review boards (see Chapter 8 for details). The respective TT would be closed via a corresponding CCR to either ratify, i.e., to make permanent the prior temporary/emergency action taken by the TTRB or to consider normal priority (scheduled) changes for incorporation into future change releases.

(3)   The EMD CM Administrator is responsible for logging the CCR into the Change Request Manager.

(4)   The CCB chair assigns an evaluator and the EMD CM Administrator coordinates impact assessment.

(5)   Class I change requests (proposed changes that affect controlled milestones, schedules, budget, cost and requirements) are forwarded to the ESDIS CCB for consideration with recommendations from the EMD Review Board.

(6)   Class II change requests (proposed changes affecting documentation, hardware [alternative use of], software [correction of errors], and COTS substitution without a Class I impact) are considered by EMD Review Board deliberations.

(7)   Notice of proposed changes is distributed to affected parties and review board members to obtain and coordinate impact assessment and optimize the approach to implement proposed changes.

(8)   The results of EMD Review Board deliberations are factored into review board resolutions which determine whether, when, or where the system changes will be implemented.

(9)   Approved changes are processed by the EMD CM Administrator to the support activities, i.e., site CCBs, support personnel (EMD), vendors, etc. who are provided with change orders, schedule, and implementation instructions.

(10)   Disapproved changes are processed by the EMD CM Administrator via official notifications, memo to the file, and update of the Change Request Manager (CRM).

(11)   The EMD CM Administrator tracks implementation and closure of CCRs via directions to implementing organizations and their acknowledgements using the CRM tracking and statusing features.

(12)   New versions and/or maintenance updates are annotated in Baseline Manager at the SMC and at the affected sites by following the procedures for configuration identification, activation dates, deactivations dates, and issuing version description documents.

(13)   Simultaneously, the SW Change Manager (ClearCase) is updated with directory trees, installation files, and software as required by SW maintenance.

(14)   Status of this activity to implement changes and assigned responsibilities is tracked through closure in the CRM at SMC and at the sites.

(15)   The databases are synchronized by manual checking between applications Baseline Manager vs. CRM vs. SW Change Manager) and automated verification by the SW CM Manager for purposes of SW distribution and maintenance.

(16)   The TT Review Board is empowered to make emergency fixes without common baseline changes and update these changes directly to Baseline Manager with documentation to follow via the CCR submitted to the appropriate CCB. Proposed common baseline changes must be submitted by CCR to the EMD Review Board.
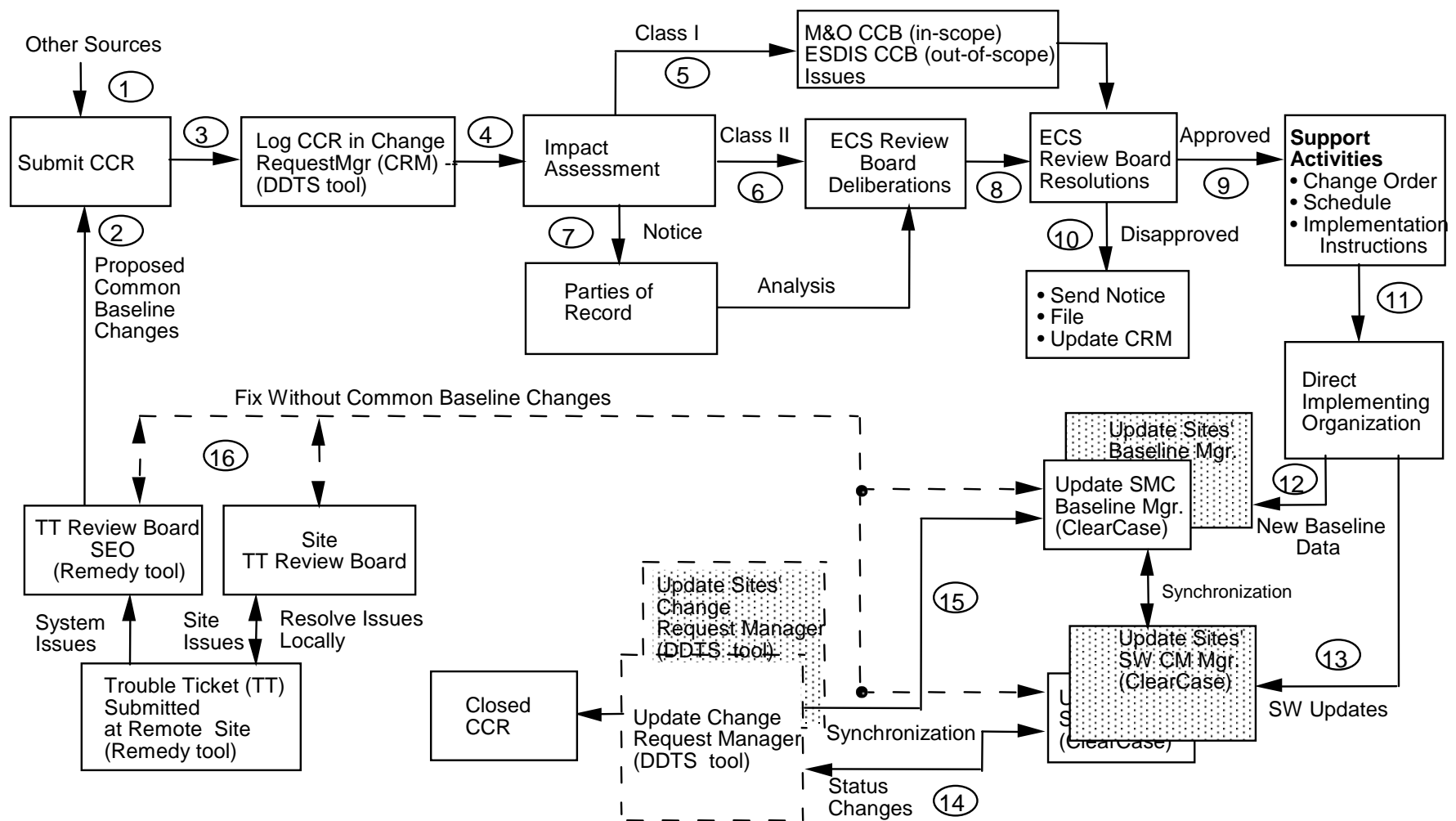
Figure 9.2.4-4. Workflow Diagram for EMD CM Administrator

Site-level Change Control Procedures

(The enumeration corresponds to the diagram of Figure 9.2.4-5)

(1)   Configuration Change Requests are received by the Site CM Administrator from all sources with regard to the **site unique extensions** to the operational EMD Core System as described in Section 9.2.4.1. These changes designated as from other sources could involve system enhancements, procedures, interfaces (both external and internal), documentation changes, etc. that are not the subject of contemporaneous problem reports which would be first deliberated by the Site/ EMD Trouble Ticket Review Board (TTRB) and/or Failure Review Board as explained below.

(2)   Proposed site baseline changes will be proposed based on Trouble Ticket (TT) resolutions obtained from the respective review boards (see Chapter 8 for details). The respective TT would be closed via a corresponding CCR to either ratify, i.e., to make permanent the prior temporary/emergency action taken by the TTRB or to consider normal priority (scheduled) changes for incorporation into future change releases.

(3)   The Site CM Administrator is responsible for logging the CCR into the Change Request Manager.

(4)   The CCB chair assigns an evaluator and the Site CM Administrator coordinates impact assessment.

(5)   Class I/System Issues change requests (proposed changes that affect controlled milestones, schedules, budget, cost and requirements) are forwarded to the EMD Review Board for consideration with recommendations from the Site CCB. Class I issues are further forwarded with recommendations by the EMD Review Board to the SCDV CCB for in-scope issues and to the ESDIS CCB for consideration of out-of scope issues with respect to the SOW of the EMD Contract.

(6)   Class II change requests (proposed changes affecting documentation, hardware [alternative use of], software [correction of errors], and COTS substitution without a Class I impact) are considered by Site CCB deliberations.

(7)   Notice of proposed changes is distributed to affected parties and review board members to obtain and coordinate impact assessment and optimize the approach to implement proposed changes.

(8)   The results of Site CCB deliberations are factored into CCB resolutions which determine whether, when, or where the system changes will be implemented.

(9)   Approved changes are processed by the Site CM Administrator to the support activities, i.e., other CCBs, support personnel vendors, etc. who are provided with change orders, schedule, and implementation instructions.

(10)   Disapproved changes are processed by the Site CM Administrator via official notifications, memo to the file, and update of the Change Request Manager (CRM).

(11)   The Site CM Administrator tracks implementation and closure of CCRs via directions to implementing organizations and their acknowledgements using the CRM tracking and statusing features.

(12)   New versions and/ or maintenance updates are annotated in Baseline Manager at the affected sites and the SMC by following the procedures for configuration identification, activation dates, deactivations dates, and issuing version description documents.

(13)   Simultaneously, the SW Change Manager (ClearCase) is updated with directory trees, installation files, and software as required by SW maintenance.

(14)   Status of this activity to implement changes and assigned responsibilities is tracked through closure in the CRM at the sites.

(15)   The databases are synchronized by manual checking between applications Baseline Manager vs. CRM vs. SW Change Manager) and automated verification by the SW CM Manager for purposes of SW distribution and maintenance.

(16)   The on-site TT Review Board is empowered to make emergency fixes without common baseline changes and update these changes directly to Baseline Manager with documentation to follow via the CCR submitted to the appropriate CCB. Proposed common baseline changes must be submitted by CCR to the EMD Review Board.

Each site's CCB accepts initial release or updates from the ESDIS CCB. Similarly, the Distributed Active Archive Center (DAAC) CCBs will accept product generation software from an ESDIS authority. Local tailoring and installation decisions are determined by the site CCB.

In the case of Evaluation Package (EP) and Prototype deliveries, the EMD CCB as directed by ESDIS will provide a configured, documented, executable with supporting files. Again, installation decisions are determined by the site CCB.

Figure 9.2.4-5. Workflow Diagram for Site-level CM Administrator

Each Science Computing Facility (SCF) is assumed to have a configuration control function. For commonality with other sites, it is assumed that this function will be performed by a CCB. A major difference, though, is the EMD contractor does not have an active role in the support of this CCB.

The SCF CCB will provide two types of configuration control:

(1)   Configuration control of software and databases that are to be executed in another site's environment. In this mode, it operates very much as the EMD CCB does to establish a product baseline.

(2)   Configuration control of SCF resources that are made available to the EOSDIS community. In this mode, its functions are the same as a DAAC CCB.

The EMD CM function at each DAAC will accept science SW and data items from the SCF CCB. These items will be incorporated into the DAAC's operational baseline as directed by the DAAC CCB.

The EOC CCB will control the operational configuration of the required EOC operational baseline. EMD CM will provide services as directed by the CCB.

The EMD Review Board will be charged with the responsibility for centralized coordination and control of EMD CM activities to ensure:

• EMD integrity and quality of service

• Successful coordination with both internal and external networks, systems, and on-site facilities

• Timely ESDIS CCB visibility into and oversight of EMD operations

• Convenient user administrative services

9.3 Configuration Status Accounting Procedures

9.3.1 Purpose

Operational phase configuration status accounting (CSA) consists of recording and reporting information about the configuration status of the EMD Project's documentation, hardware and software throughout the Project life cycle. Periodic and ad hoc reports keep ESDIS informed of configuration status as the operational mission evolves. Reports to support reviews and audits will be extracted as needed starting from the RRR.

The Baseline Manager tool described in Section 9.7 records and tracks as-built products designated as EMD control items (i.e., custom, COTS, science, toolkit, (etc.) SW and HW items along with their associated documentation and records) and historical versions of EMD operational configurations. Baseline Manager, which is updated with the acceptance tested version of the EMD baseline at RRR, records and reports EMD document change status and histories, mission milestone baselines, and change status.

CSA entails maintaining version histories of delivered and maintained products as well as histories of operational baselines and changes made to each baseline. Additionally, CSA tracks the status of proposed changes from initial CCR submission to ultimate disposition and/or implementation. CSA also maintains historical records of CCRs.

9.3.2 Applicability

All ESDIS chartered CCBs

9.3.3 References

ESDIS CM Plan

EMD Configuration Management Plan   110-EMD-001

9.3.4 Procedures

The following are topical items subject to periodic or ad hoc reporting on behalf of the respective CCB or a system-level summary of information that will be reported by the CM Administrator representing the operational baseline for all the EMD sites.

(a) **New CCRs and Revisions.** This is a standard Change Request Manager report. This report will be issued monthly and summarized annually.

(b) **CCB Review.** Distribute CCR copies for review (and Impact Analysis forms if applicable). Print the agenda and distribute prior to the meeting.

(c) **Open Action Items.** Open action items should be statused regularly between meetings.

(d) **CCB Meeting.** Record the CCB's disposition of each CCR.

(e) **Record Action Items.** Record actions, assignments, and due dates.

(f) **CM Librarian Maintained Document Changes.** When all authorized document changes have been accomplished prepare DCN, post the final version on the EMD Document Data Server and distribute hardcopy as required.

(g) **Minutes Distribution.** Distribute minutes to the standard distribution and inform actionees of assigned action items.

(h) **CCR Implementation Status.**

- After CCB disposition, regularly status open CCRs until closure.

- Class I events include: CCR to EMD Review Board for review/approval; Technical Review Board; and ESDIS Disposition

- Further events are as follows for implementation status: Consent Obtained; Item Received; Installed; Document Completed; etc.

- CCR CLOSED: A Class I CCR is not closed until the ESDIS contract officer's authorization is received or the reference CCR has been withdrawn.

- Class II document change CCRs may be closed with the CM Administrator's issuance of the DCN.

- Other non-document change CCRs may be closed when the originator verifies to the CM Administrator that all specified changes have been implemented.

9.4 Configuration Audits

9.4.1 Purpose

EMD will support Functional Configuration Audit /Physical Configuration Audit (FCA/PCA) by IATO at RRR. EMD will also support audits by ESDIS and our own Quality Office functions. Internal CM self-audits will be conducted by EMD. Self-audits evaluate the Project's compliance with the Configuration Management Plan for the EMD Project and the ESDIS CMP. The CM self-audits will verify:

- That CM policies, procedures, and practices are being followed.

- That approved changes to documentation, and to software and hardware products are properly implemented.

- That the baseline documentation of each CI agrees with the as-deployed configuration and that adequate records of differences are available at all times.

A post-audit report is written outlining the specific items audited, audit findings, and corrective actions to be taken. All action items are tracked to closure.

In addition, EMD supports formal audits scheduled and conducted by ESDIS. These audits are conducted to validate that each EMD CI is in conformance with its functional and performance requirements defined in the technical documentation. The audits validate that:

- The as-built configuration compares directly with the documented configuration identification represented by the detailed CI specifications.

- Test results verify that each EMD product meets its specified performance requirements to the extent determinable by testing.

- The as-built configuration being shipped compares with the final tested configuration. Any differences between the audited configuration and the final tested configuration are documented.

- When not verified by test, the compatibility of EMD products with interfacing products or equipment is established by comparison of documentation with the interface specifications that apply.

- COTS products are included in audits as integral parts of the EMD baseline.

9.4.2 Applicability

All ESDIS chartered CCBs

9.4.3 References

Configuration Management Plan for the EMD Project   110-EMD-001

9.4.4 Procedures

The audits are standardized for a limited set of issues that drive the process for which the audit is taken, viz., FCA/PCA, Security Issues, General Accounting, Test Readiness Review, or Operational Certifications. The baseline references for the audits will be maintained by the ClearCase Baseline Manager tool (cf. Section 9.7). Release Notes documents will be used to document auditable changes to configured articles that are issued at the EMD configuration item (CI) level. Audit processes, including Project Instruction documentation, are discussed in the *Configuration Management Plan for the EMD Project*, 110-EMD-001.

The release notes will contain the prioritized current status summary of any Trouble Tickets/Discrepancy Reports against the CI that is being issued per the change request.

Some general guidelines and/or items that must be tailored for the specific size and scope of configuration audit to be conducted include:

(a) Audit Plan;

(b) Conference Agenda;

(c) Location to collect and analyze data; conduct meetings;

(d) Applicable specifications, drawings, manuals, schedules, design and test data;

(e) Test Results Analysis;

(f) Meeting minutes including resulting audit action items;

(g) Tools and inspection equipment necessary for evaluation and verification;

(h) Unencumbered access to the areas and facilities of incoming inspection, fabrication, production, and testing;

(i) Personnel from each engineering, production, and quality department to be available for discussion of their respective areas;

(j) Copies of inspection reports, process sheets, data sheets, and other documentation deemed necessary by the Government FCA/ PCA teams; and

(k) Isolation of the item(s) and detailed parts to be reviewed.

9.5 Archiving Procedures for the SW CM Manager (ClearCase)

9.5.1 Purpose

This section details the concerns and procedures used by Configuration Management for the backup of ClearCase Version Object Base (VOB) data.

9.5.2 Applicability

All EMD CM Administrators, SW Maintenance Engineers, and Sustaining Engineers

9.5.3 References

EMD Configuration Management Plan   110-EMD-001

9.5.4 Definitions

<u>Build</u> - an assemblage of threads to produce a gradual buildup of system capabilities.

<u>EMD Development Facility (EDF)</u> - the software development environment including data, hardware, software, networks, facilities and procedures used to support EMD software development and testing.

<u>Software</u> - for the purposes of this instruction, software includes all EMD-developed application software, COTS software, build and environmental instructions, and databases used in the execution of these products.

<u>System-level</u> - for the purpose of this instruction, system-level includes all EMD integration and test activities beginning with installation of software.

<u>Software Development File</u> - a repository for a collection of material pertinent to the development or support of software.

<u>Thread</u> - a set of components (software, hardware, and data) and operational procedures that implement a scenario, portion of a scenario, or multiple scenarios.

<u>View</u> - a unique workspace management or "sandbox management" that provides developers with transparent, file-level access to any version of any element through the use of dynamically-evaluated, user-specified version selection rules.

<u>VOB</u> - Versioned Object Base. Secure, permanent, virtual file system that is mountable. Repository for storage of version-controlled data.

9.5.5 General

1. IT Infrastructure maintains a backup for all EMD systems. This procedure documents the additional steps CM has taken to ensure that the Infrastructure backups include the nightly backup of the ClearCase Versioned Object Base (VOB) Data. These VOBs store all versions of all of all custom software developed for the EMD project. In the event of a catastrophic failure, everything can be restored from the VOB data.

2. There is a cron job that runs on each VOB server (sweat, tears, nelix, janeway and triton) at 11:55 pm as the user id "vobadm" to lock down the individual VOBs (necessary to flush the database caches back to disk, in order to obtain a valid file for backup), tar up important VOB storage directories (the entire source pool, and the local copies of the VOB storage directories, which includes the VOB databases), and copy that data across

the network to a staging area (located in /tools/ccbackup) along with log files of the backup process. This process takes approximately 55 minutes.

3. The staging area is then backed up by IT Infrastructure using their standard backup procedures. IT Infrastructure maintains it's own process of offsite storage for the EMD systems backups.

9.5.6 Procedures

No additional procedures are needed on Configuration Management's part.

9.6 Software Delivery and Installation

9.6.1 Purpose

This section describes the delivery of Sustaining Engineer Organization-developed software from the Landover facility to a remote site (a DAAC), and subsequent installation of the software onto target hosts, in accordance with configuration management controlled processes. The process begins with an approved CCR. The CCR precisely defines the software to be released to the remote sites. Software is prepared and delivered using the *DeliveryTool*. The *DeliveryTool* is a custom ClearCase tool that was developed to ensure accurate and controlled release of software. Software installation is controlled by each sites' Configuration Change Board (CCB). Landover CSG prepares and releases the software to the DAACs, but the DAACs control the installation of the software into their operational modes.

9.6.2 Applicability

All EMD sites' Sustaining Engineers, System Administrators, CM Administrators, and Maintenance Engineers

9.6.3 References

Configuration Management Plan for the EMD Project   110-EMD-001

COTS and Custom Software Preparation and Delivery   CM-1-032-1

9.6.4 Procedures

9.6.4.1 Overview

This section describes the release of CM controlled software under the authority of the SCDV CCB's approved CCRs.

Assumptions:

- CM maintains records of software preparations and deliveries.

- Baseline records are maintained in the Baseline Manager (ClearCase tool)

- Electronic Delivery is performed via Secure Copy (scp) to target DAACs.

- Resource Planning, Mode Management, and other issues are not addressed in this scenario.

Summary of Procedures:

- Software is prepared delivered using a CM operated *DeliveryTool*. An approved CCR specifies which previously prepared software is to delivered and to which sites. Refer to CM-1-032-1, *COTS and Custom Software Preparation and Delivery*

- DAAC CCB Approves the Installation of the delivered software into a DAAC Operational Baseline (e.g., a mode, "OPS", "TS1", etc.)

- Software is installed in accordance with CCR installation instructions, and in accordance with local DAAC procedures/processes

- Delivery Tracking pages are maintained by EMD CM, located at:

http://pete.hitc.com/baseline/

LPDAAC	http://e0ins01.edcb.ecs.nasa.gov:10160/baseline/
GES	http://g0ins01.gsfcb.ecs.nasa.gov:10160/baseline/
LaRC	http://l0ins01.larcb.ecs.nasa.gov:10160/baseline/
NSIDC	http://n0ins02u.ecs.nasa.gov:10160/baseline/
SMC	http://m0mss04.ecs.nasa.gov:10160/baseline/

9.6.4.2 Functional Roles

EMD CM Administrator--Ensures that changes to the hardware, software, and procedures are properly documented and coordinated. Recommends configuration of all EMD configured hardware and software.

DAAC/SMC CM Administrator--Ensures that changes to the hardware, software, and procedures are properly documented and coordinated. Maintains control of all configured hardware and software. Assists in the development and administration of the library with respect to local configuration management procedures.

DAAC Sustaining Engineering--SW Maintainer--Produces, delivers, and documents the corrections, modifications, and enhancements made to EMD software (including COTS), and/or adaptations or incorporations of EMD COTS software.

9.7   Baseline Manager

9.7.1   Overview

The EMD provides a *ClearCase* Baseline Manager (BLM) tool to assist in documenting changes to the baseline, and to maintain a historical record of those changes. The BLM tool is used at the EMD Development Facility to maintain system-level records and site-level records; baseline reports are accessible at the operational sites by accessing the URL http://cmdm-ldo.raytheon.com/baseline/. This is the EMD Baseline Information System.

Selection of the "Technical Documents" button provides detailed reports regarding the EMD baseline.

Baseline Manager (BLM) is used to record and report the CCR approved operational baseline of EMD products. It contains the configuration records for each product baseline. It identifies products by CI name, description, location, version, and component configured articles. It also provides traceability to previous configurations. The BLM tool is an extension of the *ClearCase* COTS software. It provides access to functions for maintaining control item and bill of material information. The baseline data may only be affected by approved CCRs.

9.7.2 Baseline Terms and Concepts

Baseline management is a process to identify and control versions of hardware and software, to provide a standard configuration of systems throughout all sites, and allow unique site-configured systems and baselines. It identifies interdependencies between hardware and software items, and permits maintenance of a complete history of baseline changes throughout the life of the project. For EMD baseline management and BLM tools, certain terms and concepts are key to understanding how data on the system baseline are stored and tracked.

Control Item –	any EMD item under version control by Configuration Management.
Configuration Item –	an aggregation of hardware, firmware, software, or any discrete component or portion, which satisfies and end user function and is designated for configuration control.
Baseline –	a configuration identification document or set of such documents formally designated by the Government at a specific time during the life cycle of a configuration item (CI).
Configured Article –	a control item reportable as part of the Configured Articles List (CAL).
CIL –	a Configuration Items List (CIL) identifies the approved set of CIs that are subject to CM requirements and procedures.
CAL –	a Configured Articles List (CAL) describes all CIs, critical item hardware and software, and supporting documentation by which the exact configuration definition of the hardware and software can be determined.

Additional terms, some of which address specific entries in the BLM tool, further define how data on the system baseline items and structure are tracked.

Assembly –	An item made up of other items.
Bill of Material –	The list of items that comprise an assembly.
Product Structure –	The parent-child pairings that define the bill of material for an assembly; each product structure record specifies the effective

	dates and quantities for a single component of a parent for each engineering change.
Active Date –	The date a component becomes effective in an assembly's bill of material. This date is the CCR approval date.
Inactive Date –	The date a component is no longer effective in an assembly's bill of material. This date is when a new CCR displaces or affects the item.
Engineering Change –	A mechanism for grouping, reporting, and controlling product changes collectively.
Revision –	The sequence number of a product structure change to an assembly; it signifies a change to the configuration of an assembly that does not alter its form, fit, or function.
Implementation Status –	A record describing the deployment of a control item to a site and the current state and associated date of its implementation; each control item has one record for each site to which it is deployed.
Exporting Data –	Creating a formatted file or records extracted from the BLM database; control item engineering change, product structure, and interdependency records may be extracted and sent to other application software at any site.
Importing Data –	Loading BLM data from a formatted file.

At the lowest level, the baseline is composed of configured articles that are the specific types of items that make up EMD and are tracked using the BLM tool. It is important to recognize, however, that we impose a conceptual structure on those configured articles to help us think about the system. In fact, it is possible to conceptualize the structure of the system in a number of different ways, and we may select a different conceptual structure based on the requirements of the situation. The EMD baseline management approach and the BLM tool permit recording and tracking these different conceptual baselines, which can be related to the same records of the configured articles.

For example, system designers may conceptualize the system in terms that will help them track subsystems and the configuration items for which each subsystem team is responsible. This may produce a baseline structured according to a design view, such as that shown in Figure 9.7.2-1.

Figure 9.7.2-1. EMD Baseline Concept from a Design (CIL/CAL) View
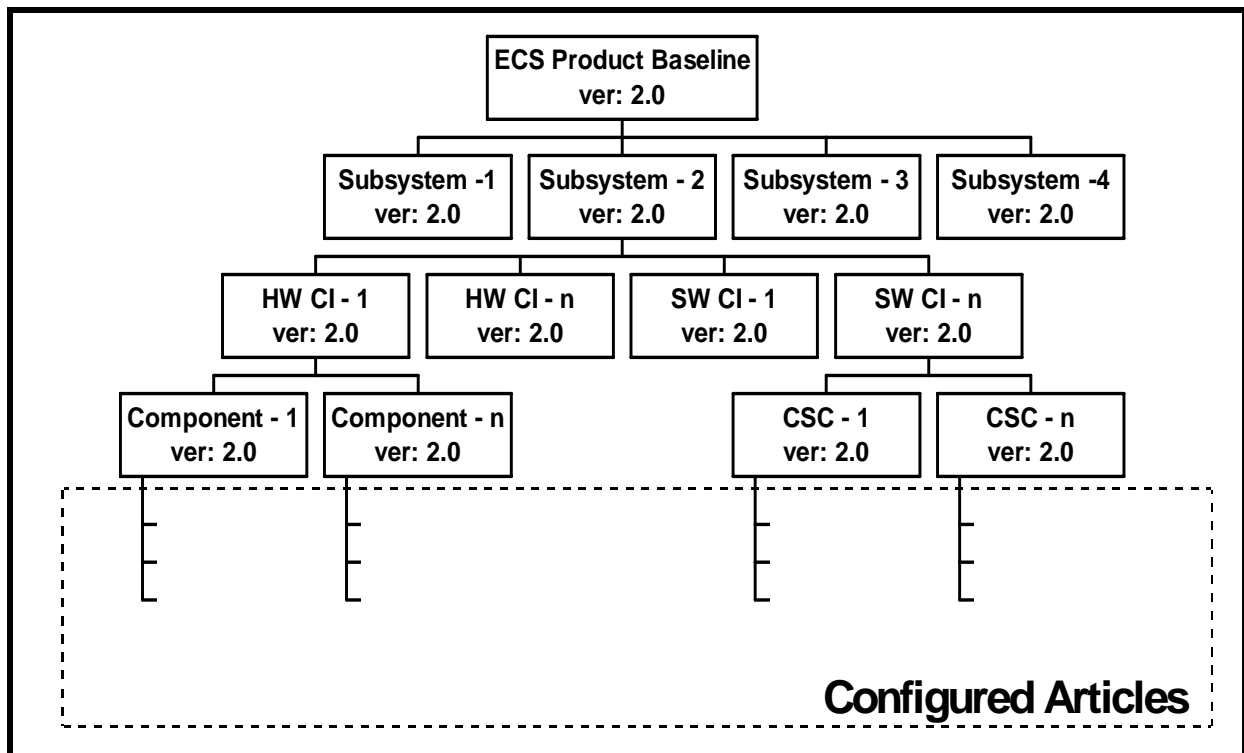
At an operations site, the concept reflected in the upper layers of the Design View baseline structure may not be particularly useful. Although the same configured articles are involved, it may be desirable, for instance, to track items from the viewpoint of network administration. The resulting baseline product structure may reflect that shown in Figure 9.7.2-2.

Even if an operations site is to view EMD product structure as composed of subsystems, it is likely that the concept of CIs will be of little use. Instead, the site is likely to be focused on what hosts make up the subsystems. Therefore, the subsystem view at an operations site may be similar to that illustrated in Figure 9.7.2-3.

The Baseline Manager database implemented at the EMD Development Facility reflects EMD-developed product structures, and site personnel may not normally need all the data necessary to define these product structures. Instead, BLM tasks are likely to be limited to areas such as noting system changes, perhaps in the context of site-unique requirements and data. However, an understanding of the different ways of conceptualizing the system will help in interpreting baseline data reflected in the BLM.
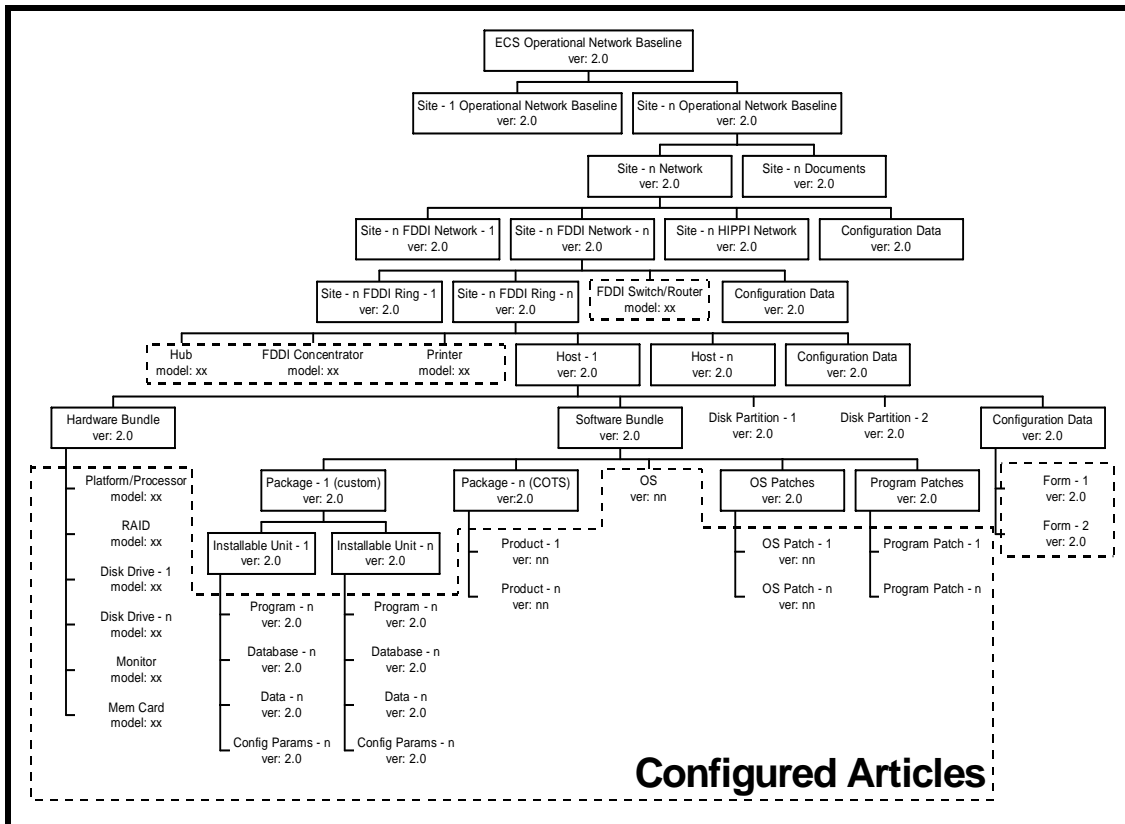
Figure 9.7.2-2. EMD Baseline Concept from an Operational (Network) View
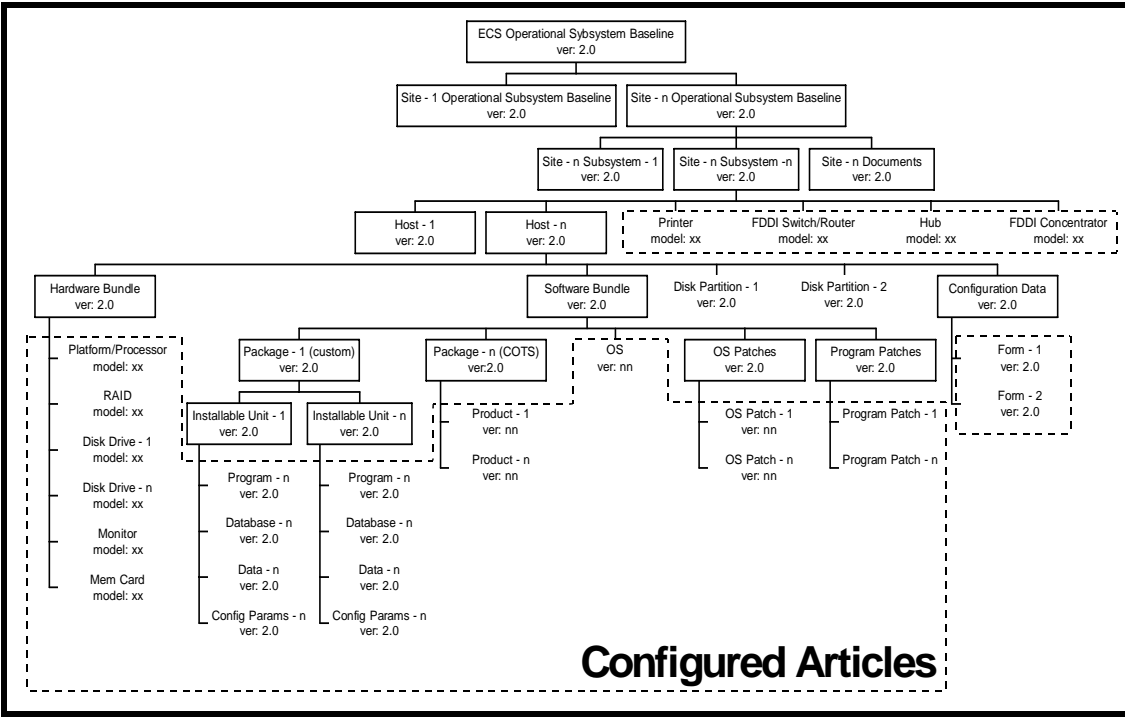
Figure 9.7.2-3. EMD Baseline Concept from an Operational (Subsystem) View

611-EMD-001, Rev. 02

9.7.3 Baseline Manager (BLM) Outputs Useful at the Sites

The BLM manages the COTS software, operating system patches, and COTS software patch baselines. The BLM records, including information on all scripts, data, and GUIs, are maintained and managed at the EMD Development Facility using ClearCase. The BLM tool produces some of the 910/920 Technical Document reports, with automated posting to EMD Baseline Information System (EBIS) and replication to the mirror site, i.e., http://cmdm-ldo.raytheon.com/baseline/. The reports include the following documents that affect all sites:

- COTS Software Versions 910-TDA-003.

- Site-Host Map 910-TDA-005.

- Critical COTS Software List 910-TDA-023.

- COTS S/W Where-Used Reports 910-TDA-030.

The reports also include the following documents that are site specific:

- Hardware-Software Maps 920-TD(x)-002.

  (*Note*: The x represents a letter designating specific DAACs. *e.g.*, g = GSFC, l = LaRC, e = LP DAAC, s = SMC, v = VATC, p = PVC, and n = NSIDC.)

- O/S and COTS S/W Patch Maps 920-TD(x)-014.

All BLM records are related to approved Configuration Change Requests (CCR) and Release Notes documents (i.e., series 914-TDA-xxx).

The Configuration Management (CM) organization is the principal user of the BLM tool, using it to implement changes to the baseline. The system is used daily to describe CCB-approved system components and to track sites and machines where version-controlled items are configuration controlled. In addition BLM supports other functions such as configuration audits, system engineering and deployment activities. The BLM records describe the hosts for each site. The sites are the operational Distributed Active Archive Centers (DAACs), Verification and Acceptance Test Center (VATC) and Performance Verification Center (PVC). The system also tracks the COTS software and patches that are mapped to their respective hosts. EBIS accommodates the identification of all configuration-controlled items such as documents, and disk partitions.

The BLM capabilities are used to:

- Maintain records that identify what items comprise individual, baselined, system configurations

- Identify the versions and variants of hardware and software items that are currently baselined together with the assemblies (e.g., hosts, subsystems, and networks) that use them

- Record item interdependencies and the sites to which baselined items are deployed

- Keep chronological histories of baseline changes and traceability of items to predecessor versions and system releases

9.7.4 Procedure for Retrieving Baseline Reports

When the EMD software baseline is changed (e.g., addition of a script, update or replacement of a Graphical User Interface (GUI) package), the change must be reflected in the collection, or "catalog," of control items that make up the affected Computer Software Component (CSC) assembly in the EMD product structure. In the BLM software at the EDF, to document the change it is necessary to add the new element to the catalog of version-controlled items, define an engineering change for the CSC assembly, and include the element in the list of items that will now make up that assembly. Once the change is documented, baseline reports reflect the new information. These reports may be accessed through the EMD CMDM server for the EMD Baseline Information System (EBIS). See Figure 9.7.4 -1 .

 The EBIS page is obtained by a click on the **EMD Baseline** button at the end of the Tools line near the top of the EMD CMDM Server page. On the EBIS page, the EMD Baseline Information Technical Documentation is accessible through use of the **Technical Documents** button at the top of the row of buttons on the left side.

The resulting **EMD Baseline Technical Documentation** page lists the document series, title, and document number. The document numbers are links that provide access to the listed documents. The titles of some documents indicate BLM origin by inclusion of the parenthetical notation (**ClearCase**).

The following procedure is applicable for retrieving baseline reports.

1. At the UNIX command shell prompt, type **setenv DISPLAY** *clientname***:0.0** and then press the **Return/Enter** key.

   - For *clientname*, use either the local terminal/workstation IP address or its machine name.

2. Start the log-in to a Netscape host by typing **/tools/bin/ssh** *hostname (e.g.*, g0ins02, e0ins02, l0ins02, n0ins02) at the UNIX command shell prompt, and press the **Return/Enter** key.

   - If you receive the message, **Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?** type **yes** ("**y**" alone does not work).

   - If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears; continue with Step 3.

   - If you have not previously set up a secure shell passphrase; go to Step 4.

3. If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, type your *Passphrase* and then press the **Return/Enter** key. Go to Step 5.

4.	At the ***<user@remotehost>*'s password:** prompt, type your ***Password*** and then press the **Return/Enter** key.

- You are logged in and a UNIX command shell prompt is displayed.

5.	Type **netscape &** and then press the **Return/Enter** key.

- The Netscape web browser is displayed.

6.	Click in the **Netsite:** field.

- The field is highlighted.

7.	Type the Universal Resource Locator (URL) for the EMD Baseline Information Server (**http://cmdm-ldo.raytheon.com/**) and then press the **Return/Enter** key.

- The **EMD CMDM Server** home page is displayed, offering access to EMD Baseline information as well as a number of tools, EMD web sites, and NASA EOS web sites.

8.	On the Tools line click on the **EMD Baseline** button.

- The **EMD Baseline Information System** page is displayed

9.	Click on the **Technical Documents** button.

- The **EMD Baseline Technical Documentation** page is displayed.

10.	Locate the desired report, scrolling down as necessary.

- Reports derived from the BLM may be indicated with a parenthetical notation **(ClearCase)** in the title entry.

11.	Click on the link for the document to be accessed.

- A directory is displayed with one or more document numbers and versions indicated as links.

12.	Click on the link for the document and version desired.

The document is displayed, and can be printed and searched.

A sample of the EBIS page is shown in Figure 9.7.4 -1

EMD Baseline Information System (EBIS)

Tools

Number of hits on this site since 11/01/98: **3 8 7 3 9**

EBIS Bulletin:

1. Added a year 2006 CCRs button.

Technical Documents	EMD Baseline Technical Documentation
Custom S/W Tracking	Detail Information on Release 7: 7.00, 7.01, 7.02, 7.03, and 7.10 Custom Code CCRs, Deliveries, and Installation Status
Engineering S/W	Engineering Software (/tools/HOTSHELF) delivery information
COTS S/W Tracking	Detail Information on COTS S/W CCRs, Deliveries, and Installation Status
Technical Directives	Engineering Technical Directives, and authorizing CCRs
Release Notes	914-TDA-XXX's, EMD Release Notes and authorizing CCRs
Pre-Ship Reviews	PSR documents- Installation instructions
2006 eCCRs	Approved 2006 CCRs (except Procurement type CCRs)
2005 eCCRs	Approved 2005 CCRs (except Procurement type CCRs)

Figure 9.7.4-1. Sample EBIS Page

10. Metadata Administration

Every science data product generated and archived by the system must be described to the system by metadata that are put into an inventory and then used to retrieve and distribute the data to users of the system. The Earth Science Data Model, described in document 420-EMD-001, Release 7 Implementation Earth Science Data Model, organizes the metadata into groups of related attributes and services to be performed on the data products. These "core" attributes are necessary to identify, interpret and perform services on granules and collections. The Data Model also provides for "product-specific" attributes (PSAs), i.e., attributes which are unique to a specific data product.

The smallest aggregation of data that is independently described and inventoried in the system is referred to as a data granule. Granules are organized into logical groupings called collections in which the granule metadata varies principally by time or location, called single-type collections.

Every collection is described by an Earth Science Data Type (ESDT) and is made known to the system by an ESDT descriptor file and associated software code that is built into the Data Server's dynamic link library (DLL) to perform the services.

Metadata administration includes creating and updating ESDTs. Collections may be modified and updated over time.   In addition, quality assurance is performed after ESDTs have been installed and granules have been generated and stored in the archives. Collection-level metadata can be updated by updating the ESDT. Granule-level metadata can be updated manually (i.e., not as a result of an operation such as subsetting, which modifies the science data content of a granule) by setting the Quality Assurance flags and explanations. Procedures for updating these flags are provided in Chapter 15, Quality Assurance.

10.1 ESDT Descriptor Files

The primary task in establishing a collection is providing the core and product-specific metadata attribute values. This is done by creating an Earth Science Data Type (ESDT) descriptor file. The descriptor file is also used to specify the data services that are available for granules that belong to the collection. These services are implemented as methods of a Dynamic Link Library (DLL) containing C++ code to accomplish each service. The descriptor file and the DLL are the means by which a collection is made known to the Science Data Server (SDSRV).

The ESDT descriptor is composed of sections, all in ODL, containing the following information:

- Collection level metadata attributes with values contained in the descriptor.
- Granule level metadata attributes whose values are supplied primarily by the Product Generation Executives (PGEs) during runtime.
- Valid values and permitted ranges for all product-specific attributes.
- List of services for all the granules in the collection and events that trigger responses throughout the system.

The services that apply to a collection are specified in the ESDT descriptor file. Product-specific services, such as subsetting or a product-specific acquire, require executable code to enact those services. This code is contained in the DLL. The DLL is written and tested by EMD sustaining engineering personnel.

After the ESDT (both descriptor file and DLL) has been generated it must be installed on the Science Data Server before the first data granule can be inserted. During this installation process, information from the ESDT Descriptor File is propagated to the Data Management Subsystem, the Subscription Server, and the Spatial Subscription Server, all of which must be operating during the ESDT installation process. The detailed procedures for ESDT installation are described in Chapter 26, Science Software Integration and Test.

10.1.1 Steps in Generating a Descriptor File

ESDTs for Distributable Product

These are the typical steps used in generating a descriptor file:

1 Identify desired collection-level metadata attributes.
   - For permanent and interim files use only the minimum attributes.
   - For distributable products identify all applicable attributes. This will involve reading appropriate documentation and interacting with the data provider.

2 Identify granule-level attributes.
   - If a sample metadata configuration file is available from the data provider, use this.

3 Check "valids" (allowable metadata values) for core attributes (write CCR if new valids are required).

4 Check PSAs (register PSAs if new).

5 Use custom built scripts and a text editor to generate the descriptor file.

6 Verify the descriptor file as outlined in Section 10.1.2.

7 Check descriptor files into ClearCase.

10.1.2 Verifying Descriptor Files

1 Run the PERL script "update.pl", following the instructions in the script prologue.
   - This script makes sure that the inventory metadata attributes are all listed as event qualifiers in the EVENT group.

2 Run the PERL script esdtQC.pl following the instructions in the script prologue.
   - This script checks for more than 30 common descriptor file errors.

3   Make any necessary corrections in response to errors issued.

4   Rerun the PERL script esdtQC.pl.

5   Repeat Steps 3 and 4 until there are no errors.

6   Run the testodl utility to ensure that there are no errors in the ODL structure for the descriptor file.

7   Make any necessary corrections in response to errors issued.

8   Rerun the testodl utility.

9   Repeat Steps 9 and 10 until there are no errors.

10.2 Preparation of Earth Science Data Types

An ESDT goes through pre-operational life cycle steps starting with an analysis of the collection's need and continuing through development and operational installation. This process involves actions by the Data Provider or User in addition to EMD. The procedures are detailed in Software Development (SD) Project Instruction SD-038 ESDT Creation, Testing, Maintenance, and Integration at http://dmserver.hitc.com/EMD_PAL/Indices/PIs/sdpi_index.html). The overall workflow is shown in Figure 10.2-1.

10.2.1   Definitions

Archive - A File Type indicating granules will be inserted to Data Server for long-term storage and acquisition for distribution.

Full - A level of metadata coverage intended for data products that are produced within the system.

Collection - A related group of data granules.

Granule - The smallest data element that is identified in the inventory tables.

Interim - A File Type indicating granules are temporarily stored in support of product generation.

Intermediate - A level of metadata coverage intended for contemporaneous data products that are not produced within the system.

Limited - A level of metadata coverage intended for heritage data products brought into the system for distribution

Minimal -A level of metadata coverage sufficient to uniquely identify a collection or granule.

Permanent - A File Type indicating static or semi-static granules that are used only as inputs in product
generation.
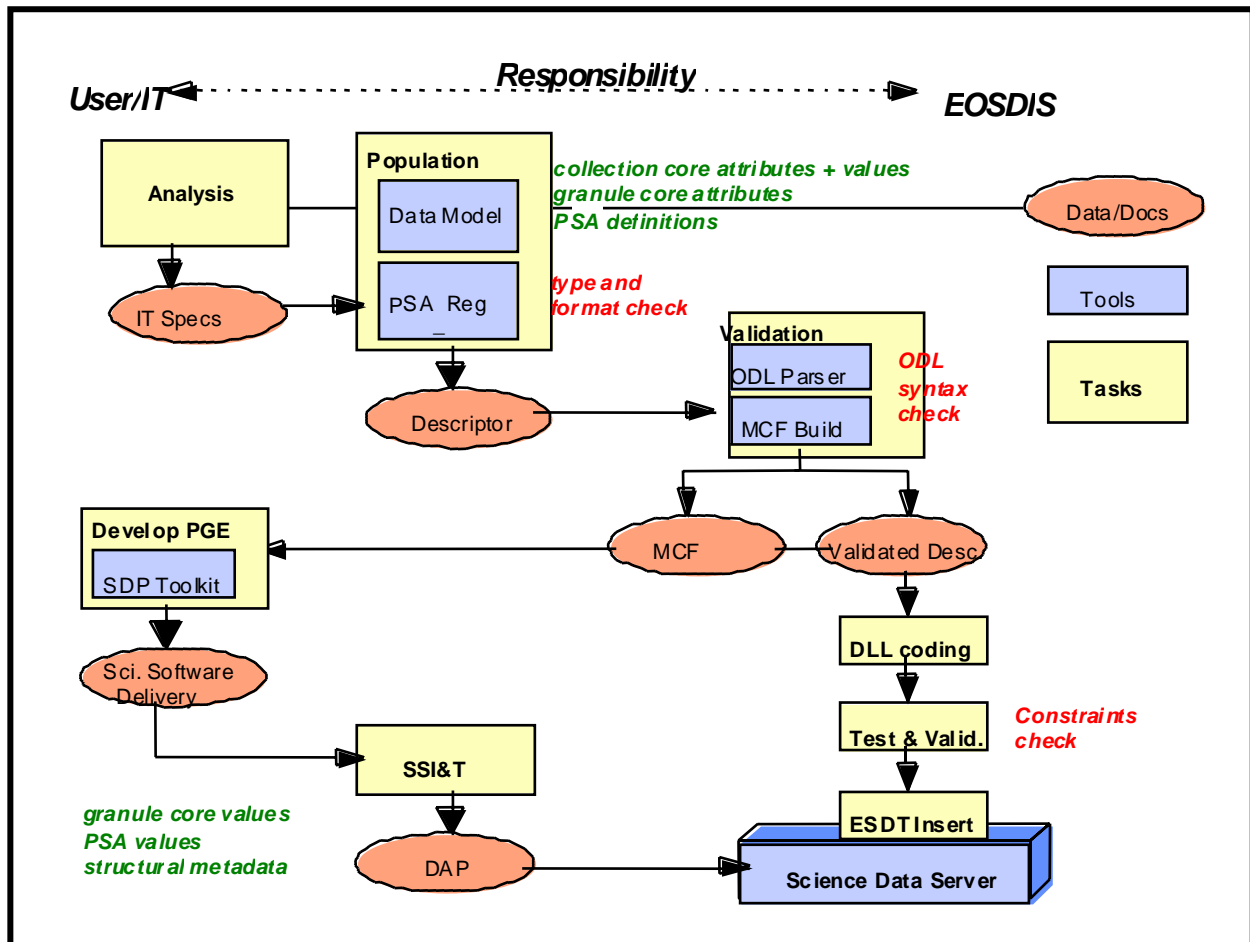
Figure 10.2-1. The ESDT Generation Process

Product Specific Attributes - Attributes that are defined by the data provider in support of searching for specific granules

Valid - An allowable metadata value.

10.2.2   Process

1   Need Analysis

- The baseline list of science ESDTs and their services is controlled by the ESDIS CCB. This baseline was established through an analysis of the system Functional and Performance Requirements Specification, the Technical Baseline established from inputs from the Ad Hoc Working Group on Production, and meetings with the individual data providers to define the basic requirements of each ESDT.
- The basic requirements are:
  - Data Provider File Designation,
  - File Type (Permanent, Interim, Archive)
  - Level of Metadata Coverage (Minimal, Limited, Intermediate, Full)
- For new ESDTs not currently in the development baseline, the result of the Need Analysis forms the basis for approving the inclusion of the ESDT into the system. This is accomplished through the CCR process.

2   ESDT Specification

- This step results in a set of specifications extending the results of the needed analysis and providing the information needed to implement an ESDT. This step is executed only if the ESDT has been included in the baseline. The roles and responsibilities for developing the specification are as above.
- The specifications must include:
  - ShortName and VersionID of the ESDT
  - A list of the metadata attributes needed, valids, and any constraints on attributes.
  - A list and specification of the services needed (e.g., specification of the INSERT, SEARCH, ACQUIRE and SUBSCRIPTION semantics).

3   ESDT Generation

- Once the ESDT Specification has been developed and the applicable attributes identified, the necessary metadata has to be gathered, the metadata values checked against the valid values and the product-specific attributes (PSA) need to be checked against the list of PSAs that are already defined (see Figure 10.2-2).
- Once the collection-level metadata and granule-level attributes have been checked, then the descriptor file is generated, the Dynamic Link Library produced (if a custom DDL is needed), and testing and validation of the ESDT performed. This process is further elaborated in the sections below.
- For a one-of-a-kind, distributable product with Full metadata coverage, this process can take up to six weeks to accomplish. For a related group of products with identical services, much of the Descriptor File and DLL of the first ESDT can be reused, and the cycle time for preparing subsequent ESDTs in the related group is much less.

Identify Attributes

Gather Information (Iterative Process)

Check Valids & PSAs

Build Descriptor File

Build DLL

Test & Validation

Quality Office Inspections

up to ~ 6 weeks
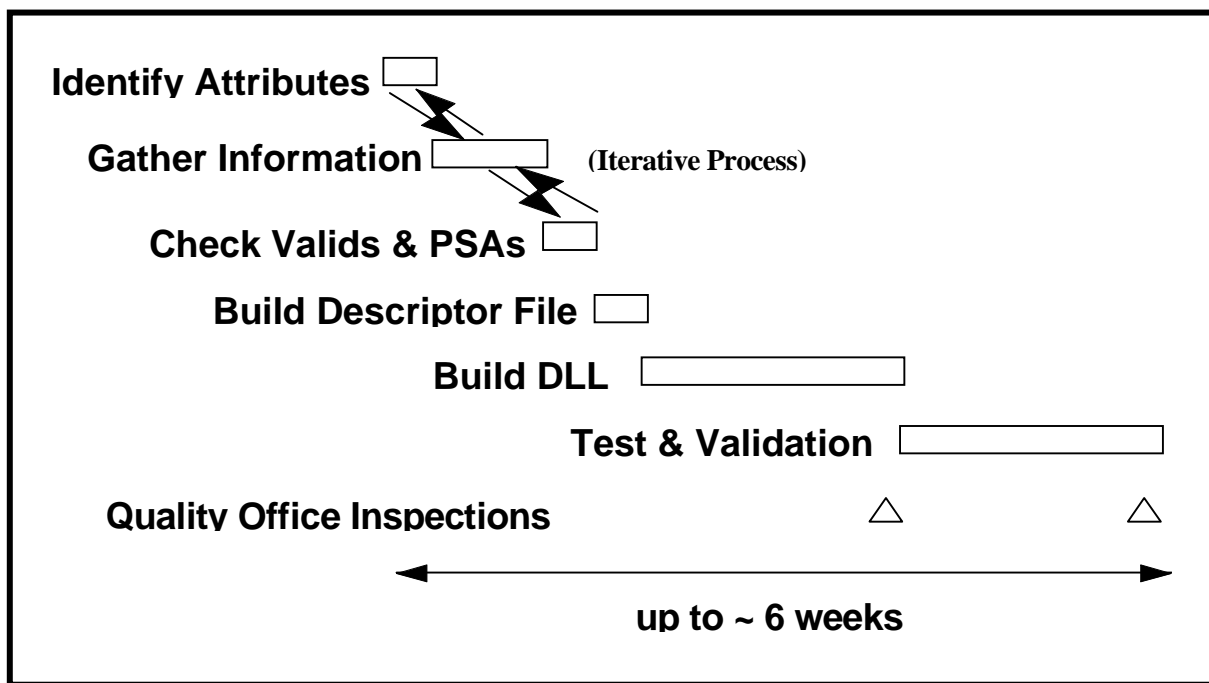
Figure 10.2-2. Steps in ESDT Development

10.3 Metadata Population

10.3.1 Collection-Level Metadata

A majority of the attributes in the Data Model apply to all the granules in the collection. These are known as collection-level attributes. There can be both core and product-specific collection-level attributes, defined once prior to establishing the collection.

Collection-level metadata is input either a text editor or a custom built script.

10.3.2 Granule-Level Metadata

The attributes in the Data Model that can vary on a granule-by-granule basis are known as granule-level attributes. There can be both core and product-specific granule-level attributes.

Granule-level metadata are specified and populated using the Metadata Configuration File (MCF). The MCF is derived from information contained in the ESDT descriptor file and is delivered by the Science Data Server for use by the Data Processing Subsystem or Ingest Subsystem. The MCF specifies how the searchable metadata attributes will be populated in the SDSRV database. For data products generated within the system, the science software or Product Generation Executive (PGE) interacts with the MCF using metadata tools contained in the Science Data Processing Toolkit. Through this process values are set for metadata attributes specified in the "source" MCF, such as the temporal or spatial coverage of each granule. These

values are then inserted into a "target" MCF at PGE run time. The MCF is used in a similar manner for data entering the system through the Ingest Subsystem.

Procedures for entering data into the system through the Ingest Subsystem are described in Chapter 16, Ingest. Procedures for running a PGE are described in Chapter 13, Production Planning and Chapter 14, Production Processing.

The actual population of the granule-level attribute values into SDSRV inventory databases takes place during the insert of a data granule into the SDSRV. Each data granule consists of one or more physical files. Accompanying each granule is a metadata record; i.e., an ASCII file containing the granule-level attributes and their values in ODL. Only one metadata record is allowed per granule, i.e., no sub-granule records are allowed, and no metadata records are shared between granules.

10.3.3   Product-Specific Metadata

Product-specific metadata can be at both the granule level and the collection level. Product-specific metadata may (at the data provider's election) be contained in the inventory tables in the database, in which case it will be searchable by the system. There is also a provision to store product-specific metadata within granules that is available only when the granule has been ordered and delivered. This is termed archive metadata and is specified in a separate ODL group in the MCF.

In the granule metadata, the core attribute that is available to store product-specific metadata is called ParameterValue. The metadata describing this attribute is specified by the data provider through the AdditionalAttributes class at the collection-level. The units of measure, range, accuracy, and resolution for this are specified in the PhysicalParameterDetails class, also at the collection-level.

Product-specific metadata at the collection level is specified at the time the other collection level metadata attributes values are defined. At the granule-level, product-specific metadata is defined in the MCF. In both cases, a list of valid values and permitted ranges are specified in the ESDT data dictionary.

10.4 Testing and Validation

Testing and validation involve installation of the ESDT on a Data Server (as described in Chapter 26), and subsequent tests of the data services for the ESDT. The tests include insertion of actual or simulated data, search, acquire, and other services that may apply and be available and supported under the extant version of the system.

After testing, the ESDT Descriptor File and DLL are promoted to the development CM environment if pre-release, or to the operational environment after the release is made operational.

This page intentionally left blank.

11. Production Rules

11.1 Introduction

Production rules are the instructions about how a particular PGE is to be run. The instructions specify a wide range of information such as the input and output data types, the frequency of execution, activation conditions and error handling instructions.

A single PGE may use one or more sets of production rules, known as PGE profiles, since it may be desirable to run the same PGE with different input granules, or activation conditions. The production rules are entered when a PGE undergoes Science Software Integration and Test (SSI&T) at the DAAC. Where applicable, default parameter values are entered at that time. The initially selected runtime parameters, metadata check parameters, and many of the default parameters can be overridden in the production environment when a Production Request is entered.

Production rules define the PGE to the Planning and Data Processing Subsystems (PDPS). The following types of conditions can be specified for each PGE:

- The time period for which the PGE will run.
    - A PGE can run every hour, every day, or for every orbit of a satellite. The frequency of how often a PGE runs must be defined to PDPS so that it knows when to plan and execute the PGE. A definition of a satellite's orbit could be included if the PGE were to be executed for some number of satellite passes.
- PGE Inputs.
    - A PGE can have any number of inputs. The types of the inputs and how frequently they are available helps determine on what basis the PGE is scheduled.
    - Most inputs to a PGE are retrieved based on time; the specified inputs are retrieved from the Data Server Subsystem for the time that the PGE is defined to execute. Production Rules allow other conditions to be added to the mix, such as checks or queries against the metadata of the input granules, or the lists of inputs as alternates (for when a primary input is not available) or optionals (for inputs without which the PGE can still run successfully). If inputs are defined as alternate or optional, the number of inputs staged for the execution of the PGE may vary from one run to the next.
- PGE Outputs.
    - A PGE can have any number of outputs. The characteristics of the outputs can have effects on any downstream PGEs that use them as inputs. For example, it is possible for an output to be defined as optional, in which case it may or may not even be produced. (When an output is not generated, it cannot be used as input for a downstream PGE.)

611-EMD-001, Rev. 02

- Runtime Parameter Values.
    – A PGE can have any number of runtime parameters, which are values that are placed in the process control file (PCF) under specified logical IDs before the PGE executes. The PGE treats them as constants and normally they are set either during SSI&T or when the Production Request is entered.
    – For some production rules (such as Orbital Processing) there is specific information that can be placed in a runtime parameter if so desired by the PGE.
    – The following two pieces of information can be placed in a runtime parameter if so desired by the PGE:
        · Data Collection Start and Stop Times. The times are specified to the PGE via two logical IDs in the Process Control File for the PGE. Logical IDs 10258 and 10259 have the Start Data Collection and Stop Data Collection times for the current DPR run of the PGE. The date/times are only accurate to the second and may not exactly match the times in the data supplied to the PGE because PDPS can match granules that are seconds (or even minutes) from the time set for the DPR.

        · Year, Month and Day of the Data Collection Start. The year/month/day information can be placed under any Logical ID that the PGE desires. They are separate entries (each goes under a different logical ID) and can provide the PGE with the **year**, **month** and the **day** of input data collection.

Some (but not all) production rules can work with other production rules.

Production rules are often used for the selection of dynamic inputs.

- **Dynamic inputs** can be either internal or external.
    – **Dynamic internal** inputs are produced by other PGEs (they are called dynamic internal inputs because they are produced at a DAAC rather than being ingested from an external data provider).
    – **Dynamic external** inputs are periodically ingested and stored in the Data Server Subsystem (they are termed dynamic external inputs because they are produced outside of the DAAC).
- **Static inputs** are granules that are inserted during the SSI&T process and are retrieved not on the basis of time but by Earth Science Data Type (ESDT) and science group. Static inputs can be chosen through the following two production rules only:
    – Metadata Query.
    – Spatial Query.

PGE profiles allow a PGE to be defined to PDPS multiple times, each with a different set of inputs, outputs, or even scheduling information. Each PGE's definition is made up of its name, its version and its profile number. Different PGE name/version pairs define different PGEs to PDPS. The addition of the profile allows for multiple definitions of a PGE name/version pair. There can be up to 99 profiles for each PGE.

11.1.1 Syntax of Production Rules

Production rules are defined in the following two ways:

- Through science metadata that is entered in various types of files during the SSI&T process.
- By entering parameter values when a Production Request is created to schedule the PGE.

During SSI&T, production rules are defined in files written in Object Description Language (ODL) in a parameter equals value format. There are three general categories of ODL files:

- PGE Science Metadata ODL Files.
- ESDT Science Metadata ODL Files.
- Production Rule-Specific Science Metadata ODL Files.
  - Orbit Definition ODL Files.
  - Path Map Definition ODL Files.
  - Tile Science Metadata ODL Files.

When a Production Request is created to schedule a PGE, it is necessary to enter certain information that is essential to implementing the production rules that affect the particular PGE. The information may concern the date and time range.

11.1.2 PGE Science Metadata ODL Files

The PGE science metadata ODL file defines a PGE (or at least the current plan for its operation) to PDPS. It specifies everything from the PGE name and version, to the period for the PGE (how often it runs), all inputs and outputs, any runtime parameters and any exit messages or dependencies. A template version of the PGE science metadata ODL file is created by the **SSIT Create ODL Template** program from a PCF from the PGE.

11.1.3 ESDT Science Metadata ODL Files

The ESDT science metadata ODL file defines a PGE input or output to PDPS. Each input and output of a PGE must have a corresponding ESDT science metadata ODL file defined. It describes everything that PDPS needs to know about the subject input or output file, from its name and version, to its period (how often data is collected), to where it is used and archived. Note that many PGEs can use the same input or output type, and thus can share the same ESDT science metadata ODL file.

Unlike the PGE science metadata ODL file, there is no tool for automatically generating a template ESDT science metadata ODL file. A template version exists under the data directory called ESDT_ODL.template. The template must be copied to a file that follows the naming convention *ESDTShortName#ESDTVersionID*.odl.

11.1.4   Production Rule-Specific Science Metadata ODL Files

The production rule-specific science metadata ODL files provide specific information to PDPS about production rules used by a PGE. They are needed only when the PGE is subject to one of the following conditions:

- Is executed on the basis of a satellite orbit.
- Needs to know the orbital path of a satellite.
- Requires data based on geographic tiling of the Earth.

Since not every PGE is based on orbits, not all PGEs require these files. The comments in the PGE_ODL.template describe when setting a specific parameter means that a production rule-specific science metadata ODL file needs to be created.

The production rule-specific science metadata ODL files are broken into two types, which are defined as follows:

- Orbit ODL File.
  - Defines the orbital period of the satellite from which the PGE's input data is created.
  - Defines when a given orbit starts, how long it lasts, and the number of the orbit.
  - PDPS uses the information in the orbit ODL file to extrapolate future orbits and is able to plan PGEs that are required to run every so many orbits of the satellite.
- Pathmap ODL File.
  - Defines the mapping between the cyclic 0-233 orbits that the satellite makes with the actual path number that the PGE requires.
  - PDPS computes the path number from the orbit number (specified in the orbit ODL file) by incrementing it until it reaches the 233 maximum, then resetting it to zero.
  - Many instruments expect the path number to be a fixed swath on the Earth, so it is not just incremented for each satellite pass.
  - The Pathmap ODL file creates a mapping from the sequential path numbers to the path numbers expected by the PGEs.

Unlike the PGE science metadata ODL file, there is no tool to automatically generate a template production rule-specific science metadata ODL file. Because the files themselves tend to be small, this is not usually a problem. A template version of each kind of production rule-specific science metadata ODL file (e.g., ORBIT_ODL.template) exists in the /usr/ecs/<MODE>/CUSTOM/data directory on the AIT Workstation. The templates must be copied, named properly, and edited in order to create the appropriate production rule-specific science metadata ODL file.

11.2 Production Rules

The following statements provide some simplified descriptions of the production rules:

- **Basic Temporal** - Temporal (time) range of inputs matches the temporal range of outputs.

- **Advanced Temporal** - Temporal range of inputs is offset from the expected temporal range of inputs and outputs.
- **Alternate Inputs** - PGE is run with different inputs based on the availability of various alternate input data sets.
- **Optional Inputs** - PGE is run with specified optional inputs if available; otherwise, PGE is run without them.
- **Minimum/Maximum Number of Granules** - Minimum number of input granules needed for full data coverage and maximum number of input granules to search for may be specified. Minimum and maximum number of outputs expected from the PGE may be specified.
- **Optional DPRs** – The only DPRs executed are those for which the non-routine key input data actually become available (i.e., are either produced in data processing or can be acquired from the archive).
- **Metadata Checks** - DPR is run only if input data's metadata value(s) meet(s) certain criteria.
- **Metadata Query** - Input granule selection is based on metadata value.
- **Spatial Query/Spatial Pad** - Input granule selection is based on the spatial coverage of another input (i.e., the key input). Spatial Pad involves adding area to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.
- **Closest Granule** – DPR is generated if a required input granule within a particular time range (rather than an exact time) is available; otherwise, no DPR is generated. (Supersedes the Most Recent Granule Production Rule)
- **Orbital Processing** - Selection of input times is based on orbit information.
- **Multiple DPRs for Insertion Time** - Allows the creation of DPRs for multiple granules with the same insertion time (affects ASTER L1B routine processing only).
- **Tiling** - Input data is chosen on the basis of Instrument Team-defined tiles (geographic areas).

11.2.1 Basic Temporal Production Rule

The Basic Temporal Production Rule defines the timeframe for the PGE along with its input and output data. PGEs subject to the Basic Temporal Production Rule generally have the following characteristics in common:

- Typically scheduled to run using input data that become available periodically (every hour, every day, etc.).
- Use input data for a particular period of time.
- Produce output for a specified length of time.

The data the PGE takes in (its input) and the data it produces (its output) have the same period (or some subset of the same period) as the PGE.

- Example One:
  - A MODIS PGE processes data for five-minute intervals, producing Level 1B granules.

- The PGE requires as input the specific five-minute Level 1A granule that is contemporaneous with (covers the same five-minute time period as) the Level 1B granule to be produced.
- Using the Basic Temporal Production Rule, a five-minute Level 1A granule is staged as input to the PGE and a five-minute Level 1B granule is expected as output, both matching the timeframe for which the PGE is run.

- Example Two:
  - A PGE for a different instrument processes data for 24-hour intervals, producing 24-hour Level 1A granules as output.
  - As input the PGE takes Level 0 data that is ingested every two hours.
  - Using the Basic Temporal Production Rule, twelve two-hour Level 0 granules are staged as input to the PGE and a 24-hour Level 1A granule is expected as output, matching the timeframe for which the PGE is run.

The fundamental elements used to define the Basic Temporal Production Rule are "period" and "boundary."

- **Period** is the length of time for which a PGE processes data or the length of time for which input and output data is collected.
  - A PGE that is subject to the Basic Temporal Production Rule only and that processes data in two-hour blocks takes in data that relates to a particular two-hour interval and produces output data for that same two-hour period.
  - Data that has a period of 15 minutes was collected or produced for a 15-minute time period.
- **Boundary** is the starting point for the data or PGE.
  - Depending on the characteristics of the data or PGE, the boundary may be the start of a minute or hour or day or week (etc.).
  - If a PGE's boundary is the start of the hour, it processes data that starts every hour and runs on data for the length of its period.
  - If data comes in every day, PDPS predicts that the data is going to be available at the start of the day and allows scheduling of PGEs that use the data as input accordingly.

Both the PGE itself and the input data have a boundary and period associated with them. That is how PDPS determines the frequency of processing for a Basic Temporal PGE and the time period for its inputs and outputs.

PDPS uses **period** and **boundary** in combination to plan the processing of each PGE, including determining its input requirements and anticipated output (which may be input to other PGEs). If a PGE has a period of one hour and a boundary of "start of day," it is scheduled every hour, beginning at midnight. If an input has a period of 15 minutes and boundary of "start of hour," PDPS predicts it every 15 minutes beginning on the hour.

Boundary offset is an addition to the Basic Temporal Production Rule that allows a PGE or data to start on an offset from a given boundary. For example, if a PGE would normally run every day but not start until two or three hours into the day (e.g., beginning at 3:00 a.m. instead of

midnight), a boundary offset can be used to add three hours to the "start of day" boundary. This would mean the PGE would run on data that occurred three hours after the boundary.

Non-routine data can be used as an input to a Basic Temporal DPR. Such data must exist at the Science Data Server for the time period of the DPR when the DPR is created or it can be retrieved using the **Closest Granule** or **Optional Input** Production Rules. If no other Production Rule is used and the non-routine data is **not** available at Science Data Server, the DPR fails at creation time. Because there is no Boundary or Period associated with non-routine data, PDPS cannot predict it; consequently, it must be available at Production Request time.

Running with data that falls within the time of the DPR only is a special case that is applicable to some PGEs. Normally, PDPS would match any granules that intersect the time of the DPR. However, some PGEs require data that falls within the time of the DPR only. For example:

- A data set is composed of granules with the following time ranges:
  - 01:00-02:00
  - 02:00-03:00
  - 03:00-04:00
- The DPR has the following time range:
  - 01:30-03:30

Usually a PGE would want all three granules in the data set because they all intersect the time of the DPR. But if a PGE wants the data that falls within the DPR time only, the 02:00-03:00 granule would be the only one selected. Restriction of the data to the time of the DPR is accomplished using the **DPR Align Flag** (ALIGN_DPR_TIME_WITH_INPUT_TIME).

The **end-of-month anomaly** is an addition to the Basic Temporal Production Rule that allows a PGE or data to cover a specific number of days within a month. The month is broken into thirds. The first third is composed of the first 10 days of the month. The second third consists of days 11 through 20. And the last third varies in length depending on the total number of days in the month (i.e., for November it would have 10 days; for December it would have 11 days). A specific **boundary** and **period** allow a PGE or its data to be scheduled into thirds of a month.

Figure 11.2.1-1 provides an illustration of the Basic Temporal Production Rule. The PGE has a boundary of "start of day" and a period of one hour, so it is scheduled for every hour through the day. If a Production Request were entered for two full days of processing, a DPR would be created for the PGE to run every hour; i.e., 48 DPRs total. If a Production Request were created for a four-hour period in the middle of a single day (for example, from 12:00 noon to 4:00 p.m.), then four DPRs would be created, one for 12:00-1:00, one for 1:00-2:00, one for 2:00-3:00, and one for 3:00-4:00.

In the example (Figure 11.2.1-1), Input One has a boundary of "start of day" and a period of two hours, so when PDPS plans for its availability, it expects a granule every two hours beginning at midnight. Consequently, each granule of Input One is associated with two DPRs for the PGE, because the PGE encompasses only one hour of the two-hour granule's period.
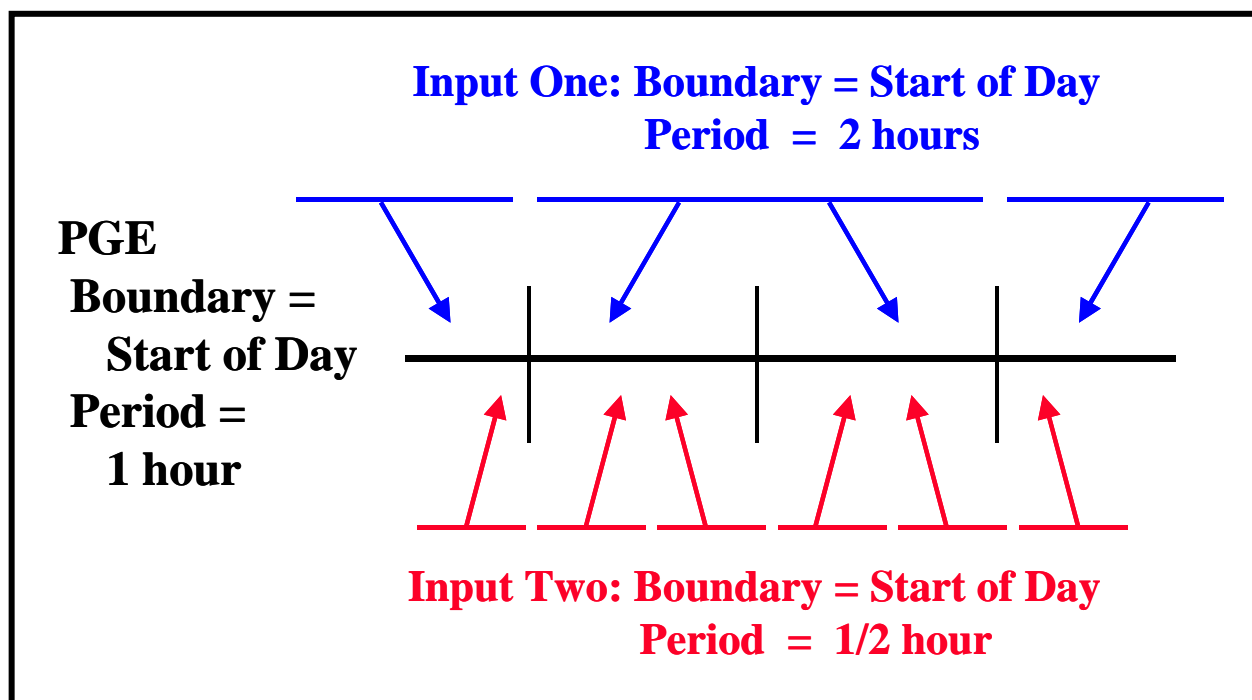
Figure 11.2.1-1. Example of the Basic Temporal Production Rule

Input Two has a boundary of "start of day" and a period of ½ hour, so when PDPS plans for its availability, it expects a granule every ½ hour beginning at midnight. As a result two granules of Input Two are associated with each DPR for the PGE, because the PGE encompasses an hour of the ½-hour granule's Period. Thus, every DPR of the PGE will wait for two granules of Input Two to arrive before it can be processed.

11.2.1.1 PGE Science Metadata ODL File Parameters

The following parameters must be set properly in the applicable PGE science metadata ODL file in order to implement the Basic Temporal Production Rule:

- SCHEDULE_TYPE.
- PROCESSING_PERIOD.
- PROCESSING_BOUNDARY.

The SCHEDULE_TYPE parameter specifies the type of scheduling that will be done for the PGE. The following values are applicable to the Basic Temporal Production Rule:

- "Time"
  - The PGE is scheduled on the basis of the specified boundary/period and the availability of data for that boundary/period.
- "Snapshot"
  - The PGE is scheduled for a single date/time.
  - Note that PROCESSING_PERIOD and PROCESSING_BOUNDARY are not needed when "Snapshot" is specified.

Other values for SCHEDULE_TYPE apply to other production rules, such as the following values:

- "Data"
  - The PGE is scheduled on the basis of the availability of data produced by other PGEs.
- "Tile"
  - The PGE is scheduled based on the definition of geographic tiles.
- "Orbit"
  - PGE scheduling is based on the orbit of the spacecraft.

The PROCESSING_PERIOD parameter describes the length of time for which the PGE executes. Data will be acquired (barring any combination of Production Rules) for the specified period and output data will be planned for the given period. It is of the format "<Period Type>=<Length of Period>". Note that "length of period" can be specified as a positive integer only. The following values are acceptable "period type" entries for the Basic Temporal Production Rule:

- "YEARS"
  - PGE processes data applicable to a given year or years.
  - "YEARS" might be specified for a PGE that computes a yearly average.
  - For example, PROCESSING_PERIOD = "YEARS=1" relates to a PGE that processes one year's worth of data.
- "MONTHS"
  - PGE processes data applicable to a particular month or several months.
  - "MONTHS" is most likely to be used for some kind of averaging PGE.
  - For example, PROCESSING_PERIOD = "MONTHS=2" relates to a PGE that processes two months' worth of data at a time.
- "THIRDS"
  - PGE processes data applicable to some number of thirds of the month.
  - For example, PROCESSING_PERIOD = "THIRDS=1" relates to a PGE that processes data applicable to 1/3 of the month.
- "WEEKS"
  - PGE processes data applicable to some number of weeks.
  - For example, PROCESSING_PERIOD = "WEEKS=2" relates to a PGE that processes two weeks' worth of data every time it runs.
- "DAYS"
  - PGE processes data applicable to some number of days.
  - For example, PROCESSING_PERIOD = "DAYS=5" relates to a PGE that processes five days' worth of data.
- "HOURS"
  - PGE processes data applicable to some number of hours.
  - For example, PROCESSING_PERIOD = "HOURS=4" relates to a PGE that processes four hours' worth of data when it is executed.

- "MINS"
  - PGE processes data applicable to some number of minutes.
  - For example, PROCESSING_PERIOD = "MINS=5" relates to a PGE that processes five minutes' worth of data.
- "SECS"
  - PGE processes data applicable to some number of seconds.
  - For example, PROCESSING_PERIOD = "SECS=2" relates to a PGE that runs on two seconds' worth of data.

There are other types of values for PROCESSING_PERIOD but they apply to other production rules (as described in the applicable sections of the lesson).

The PROCESSING_BOUNDARY parameter specifies the boundary (starting point in time) of the PGE. It tells when each instance of the PGE should start. Note that the PROCESSING_BOUNDARY and PROCESSING_PERIOD are used in conjunction to schedule the PGE.

The following PROCESSING_BOUNDARY values are used for implementing the Basic Temporal Production Rule:

- "START_OF_HOUR" – PGE processes data for each hourly interval.
- "START_OF_6HOUR" - PGE processes data for every 6-hour interval.
- "START_OF_DAY" - PGE processes data for every daily interval.
- "START_OF_WEEK" - PGE processes data for every weekly interval.
- "START_OF_ONE_THIRD_MONTH" - PGE processes data for every 1/3 of a month.
- "START_OF_MONTH" - PGE processes data for every monthly interval.
- "START_OF_YEAR" - PGE processes data for every yearly interval.
- "START_OF_TIME" - PGE runs from the specified date/time in the form "START_OF_TIME=DD/MM/YYYY HH:MM:SS".
  - Can be used in conjunction with all types of PROCESSING_PERIOD.

There are other values for PROCESSING_BOUNDARY that apply to other production rules (as described in the applicable sections of the lesson).

11.2.1.2 Retrieving Data that Falls within the DPR Time Only

When the ALIGN_DPR_TIME_WITH_INPUT_TIME flag is set to "Y" (i.e., ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y") PDPS uses only the granules for the input that fall within (instead of intersecting) the time of the DPR. If the flag is **not** set or is set to "N", granules for the input will be chosen on the basis of whether or not they intersect the time of the DPR.

11.2.1.3 ESDT Science Metadata ODL File Parameters

The following parameters must be set properly in the applicable ESDT science metadata ODL file in order to implement the Basic Temporal Production Rule:

- DYNAMIC_FLAG.
- PERIOD.
- BOUNDARY.

The DYNAMIC_FLAG describes the type of data that is defined in the ESDT science metadata ODL file. It specifies to PDPS what kind of data the PGE requires as input or produces as output. It can have any of the following four possible values, all of which are valid for Basic Temporal data:

- "S"
    - Static Data.
    - Data do not change at regular intervals.
    - The same granule can be used as input for many runs of the PGE.
    - Calibration files are a good example of static data.
- "I"
    - Dynamic Internal.
    - Data are produced by a PGE running at the local DAAC.
    - All output products are either "dynamic internal" or "interim" kinds of data.
- "E"
    - Dynamic External.
    - Data are produced by an external source (not a PGE running at the local DAAC).
    - Earth Observing System (EOS) Data and Operations System (EDOS) data is a primary example.
    - Dynamic external can be set for PGE inputs only.
- "T"
    - Interim/Intermediate.
    - Data are stored only temporarily by the Data Server Subsystem.

The PERIOD parameter specifies the length of time covered by the data. Data are expected to be either ingested or produced for the length of the PROCESSING_PERIOD described in PGE science metadata ODL files. However, the PERIOD of the data does **not** have to match the PROCESSING_PERIOD defined for the PGE. PDPS plans for data where the ESDT period is less or more than the processing period of the PGE that uses it. For example, if the PGE PROCESSING_PERIOD = "HOURS=1" and the input data PERIOD = "MINS=5", then PDPS plans to acquire twelve granules of the input data to cover the PROCESSING_PERIOD.

The following "period type" values are used for implementing the Basic Temporal Production Rule:

- "YEARS"
    - Data span a year or years.
    - "YEARS" might be selected for a yearly average output product.
    - For example, PERIOD = "YEARS=1" specifies data that cover a period of a year.

- "MONTHS"
  - Data span a month or several months.
  - "MONTHS" is most likely used for some kind of averaging output product.
  - For example, PERIOD = "MONTHS=2" specifies data that cover a period of two months.
- "THIRDS"
  - Data span some number of thirds of a month.
  - For example, PERIOD = "THIRDS=1" specifies data that cover a period of 1/3 month.
- "WEEKS"
  - Data span some number of weeks.
  - For example, PERIOD = "WEEKS=2" specifies data that cover a period of two weeks.
- "DAYS"
  - Data span some number of days.
  - For example, PERIOD = "DAYS=5" specifies data that cover a period of five days.
- "HOURS"
  - Data span some number of hours.
  - For example, PERIOD = "HOURS=4" specifies data that cover a period of four hours.
- "MINS"
  - Data span some number of minutes.
  - For example, PERIOD = "MINS=5" specifies data that cover a period of five minutes.
- "SECS"
  - Data span some number of seconds.
  - For example, PERIOD = "SECS=2" specifies data that cover a period of two seconds.
- "ORBITS"
  - Data span some number of orbits of the spacecraft.
  - For example, PERIOD = "ORBITS=1" specifies data that cover one orbit.
  - A PGE can be time-scheduled (using the Basic Temporal Production Rule) but use orbit-based data.

The BOUNDARY parameter is the starting point in time of the data granule. It tells when each data granule should start. Note that the BOUNDARY and PERIOD are used in conjunction to determine the starting and ending time for the granules.

The following values for BOUNDARY apply to the Basic Temporal Production Rule:
- "START_OF_HOUR"
  - Data granules start every hour.
- "START_OF_6HOUR"
  - Data granules start every six hours.

- "START_OF_DAY"
  - Data granules start every day.
- "START_OF_WEEK"
  - Data granules start every week.
- "START_OF_ONE_THIRD_MONTH"
  - Data granules start every 1/3 of a month.
- "START_OF_MONTH"
  - Data granules start every month.
- "START_OF_YEAR"
  - Data granules start every year.
- "START_OF_ORBIT"
  - Data granules start every orbit.

The PREDICTION_METHOD parameter specifies the method of prediction that the Planning and Data Processing Systems use when creating DPRs that use the ESDT. The parameter has two possible values:

- "ROUTINE"
  - Data granules come in at regular intervals.
  - Boundary and Period are used when predicting the inputs to DPRs.
- "NONROUTINE"
  - Data granules come in at odd or random intervals.
  - DPRs that use the input can work off data that exists in the Science Data Server when the production request is created. This is because PDPS cannot predict the data.
  - When PREDICTION_METHOD = NONROUTINE, Boundary and Period are **not** used.

11.2.2 Advanced Temporal Production Rule

The Advanced Temporal Production Rule allows for input data to be acquired for a time period other than that of the PGE or its planned inputs/outputs. It provides an offset mechanism, specifying on an input basis that the data required for processing is some number of seconds earlier or later than the planned time period for the PGE.

- Example One:
  - A PGE requires data from its previous execution for interpolation purposes (e.g., one of its inputs is the output of the very same PGE the last time that it ran).
  - If the PGE processes data for each one-hour interval (producing an hourly product), the Advanced Temporal Production Rule is specified with an offset of minus 3600 seconds (one hour) for the input of the ESDT produced by previous runs.
- Example Two:
  - A PGE takes as input two-hour Level 0 data to produce an L1A product.
  - Because the edges of the Level 0 data can be difficult to process without preceding and succeeding data, the PGE requires three Level 0 granules, one from

the time period before it runs, one for the time period it is currently processing and one for the next time period.

− The PGE is defined as having three inputs, the first with an Advanced Temporal offset of minus 7200 seconds (two hours), the second with no Advanced Temporal offset and the third with an Advanced Temporal offset of plus 7200 seconds (two hours).

The Advanced Temporal Production Rule uses the times specified in the Basic Temporal Production Rule as a reference point for specifying offset(s) to request data from a "period" and/or "boundary" different from that of the DPR or its input. The offsets are specified as either negative or positive numbers to indicate whether the time period of the input data is before or after that of the DPR (a particular run of a PGE).

- **Begin Period Offset** is an amount of time (in seconds) that is specified with respect to the DPR start time. A negative beginning offset requests data that was collected before the DPR start time. A positive beginning offset requests data with a collection time after the start time of the DPR.

- **End Period Offset** is an amount of time (in seconds) that is specified with respect to the DPR end time. A negative ending offset requests data that ended collection before the DPR end time was reached. A positive ending offset requests data that ended collection after the end time of the DPR boundaries.

Note that the beginning and ending offsets are not absolute cut-offs for data. Overlapping granules (granules that start or end outside of the offsets) will be staged as inputs to the DPR.

Figure 11.2.2-1 provides an illustration of the Advanced Temporal Production Rule. The PGE shown in the example processes data for every one-hour interval. However, Input One comes in at two-hour intervals and Input Two is produced every 1/2 hour.

Both the Begin Period Offset and End Period Offset for Input One are -7200 seconds (minus two hours). Consequently, every DPR will stage the "previous" Input One. This could be used to get the "previous" or "next" granule of an input.

The Begin Period Offset for Input Two is zero, meaning that it will match the Start Time of the DPR. The End Period Offset is +1800 seconds (plus 1/2 hour). Therefore, all Input Two granules that fall within the time period of the DPR plus 1/2 hour would be staged. The effect is to acquire all Input Two granules within the time period of the DPR, plus the one from the next 1/2-hour time period, for a total of three granules. The additional granule acquired by means of the End Period Offset might be used for interpolation purposes at the end point.

The same types of parameter settings that apply to the Basic Temporal Production Rule apply to the Advanced Temporal Production Rule. In addition, there are some parameters in the PGE science metadata ODL file that apply specifically to the Advanced Temporal Production Rule. However, the values applicable to the Basic Temporal Production Rule must be set before the Advanced Temporal Production Rule syntax is added.
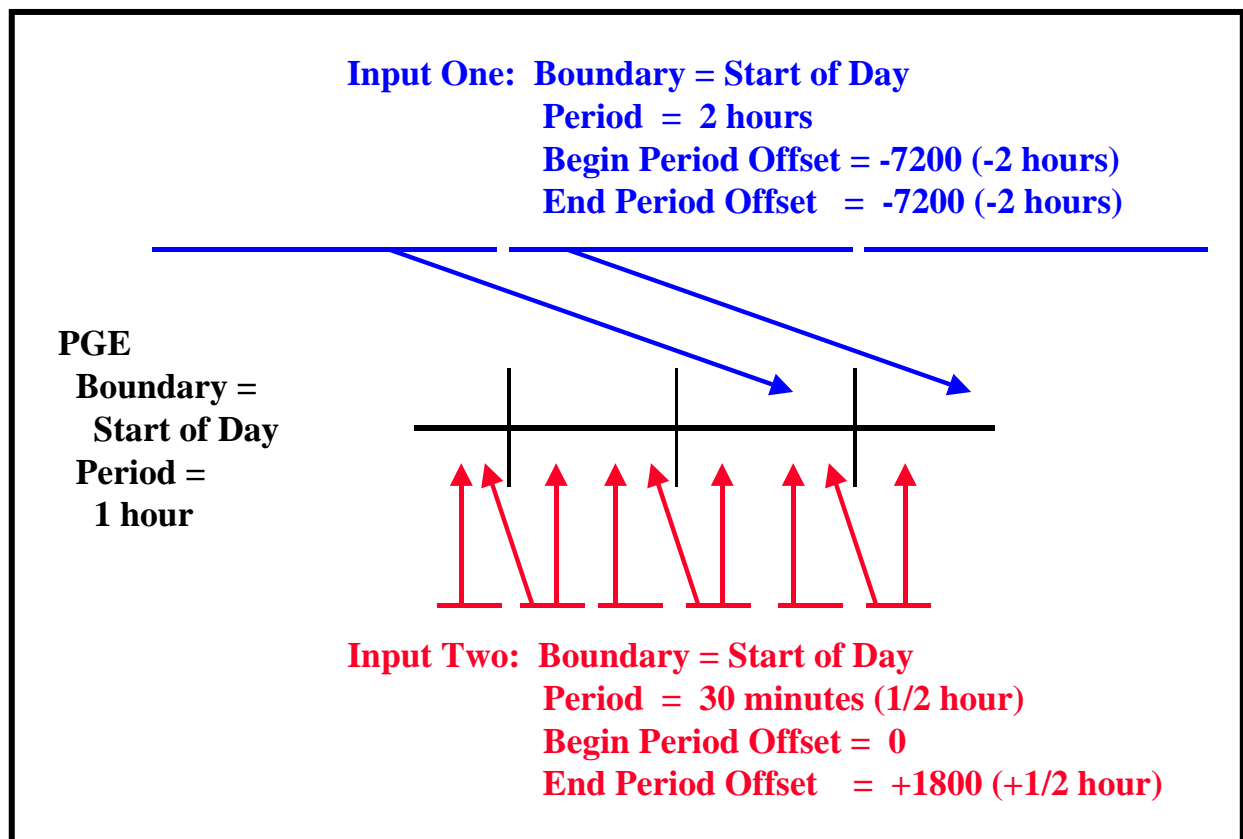
Figure 11.2.2-1. Example of the Advanced Temporal Production Rule

11.2.2.1 PGE Science Metadata ODL File Parameters

During the SSI&T process the PGE science metadata ODL file is generated from the PCF delivered with the science algorithm. A PCF_ENTRY object is generated for each file entry in the PCF. In order to implement the Advanced Temporal Production Rule the PCF_ENTRY object for each type of input file to which the rule applies uses the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  BEGIN_PERIOD_OFFSET =
  END_PERIOD_OFFSET =
  .
  .
  .
END_OBJECT = PCF_ENTRY
```

Accordingly, the following parameters must be set properly in order to implement the Advanced Temporal Production Rule:

- BEGIN_PERIOD_OFFSET.
- END_PERIOD_OFFSET.

BEGIN_PERIOD_OFFSET is the offset added to or subtracted from the Data Start Time of the DPR. The value assigned to BEGIN_PERIOD_OFFSET can be either a positive or negative value, specified in seconds. If the value is positive, it is added to the Data Collection Start Time (looking for the input forward in time). If the value is negative, it is subtracted from the Data Collection Start Time (looking backward in time). For example, BEGIN_PERIOD_OFFSET = -3600 requests data that was collected one hour (3600 seconds) before the DPR start time.

END_PERIOD_OFFSET is the offset added to or subtracted from the Data Collection End Time of the DPR. The value assigned to END_PERIOD_OFFSET can be either a positive or negative value, specified in seconds. If the value is positive, it is added to the Data Collection End Time (looking for the input forward in time). If the value is negative, it is subtracted from the Data Collection End Time (looking backward in time). For example, END_PERIOD_OFFSET = +2700 requests data that was collected 45 minutes (2700 seconds) after the DPR end time.

The BEGIN_PERIOD_OFFSET and END_PERIOD_OFFSET parameters can be specified for any input PCF_ENTRY in the PGE science metadata ODL file. If not specified, the parameters are set to zero (0) and the Advanced Temporal Production Rule does not apply to the PGE.

11.2.3 Alternate Input and Optional Input Production Rules

The Alternate Input and Optional Input Production Rules are very similar and use much the same processing in PDPS. Both rules allow a PGE to select various inputs based on timers and priority lists. The major difference is that Alternate Inputs requires that one of alternates on the list be used, whereas Optional Inputs allows successful execution of the PGE if no optional input on the list is available.

The Alternate Input Production Rule allows for a PGE to evaluate a list of inputs in priority order and be scheduled and executed with the best priority input that could be found. In essence, a PGE using Alternate Inputs is saying "I would like to run with Input A, but if it's not available, I am willing to run with Input B." A timer can be used to specify how long to wait for a given alternate choice before proceeding with a choice of lesser priority. The PGE is not executed until one of the alternate choices has been found.

- Example:
  - A PGE requires model wind data as an input but is capable of accepting wind data from a Data Assimilation Office (DAO) model, a National Centers for Environmental Prediction (NCEP) model, or (as a last resort) climatology.
  - The PGE would use the Alternate Input Production Rule to list each input in priority order, giving a timer value for how long to wait before trying the next input.
  - If the DAO data were most desirable, DAO would be listed as first choice or "primary" data.

- NCEP would be the second choice.
- Climatology would be the last choice.
- If a timer value is specified for DAO data, the PGE will wait for that timer to expire before running with either NCEP data or climatology.
- If a timer had been placed on the NCEP input, the PGE would wait before running with the climatology data.

The Optional Input Production Rule allows for a PGE to list inputs that are desired but not required for it to execute. The inputs are ranked as previously stated and timers are set to wait before choosing a lower-priority type of input. However, if none of the inputs on the list becomes available, the PGE starts because the alternatives are classified as "optional." In essence the PGE is saying "I would like to run with Input A, but if its not available, I can run (and produce reasonable output) without it."

- Example:
  - It would be preferable to run a particular MODIS PGE with the output of a MISR PGE as input.
  - However, the MISR output may not be produced every day.
  - So the MODIS PGE lists the MISR input as optional with a two-hour timer.
  - On those occasions when no MISR output is produced, the MODIS PGE waits for two hours and then is executed without the MISR input.

Figure 11.2.3-1 provides an illustration of the Alternate Input Production Rule. The PGE in the illustration has two inputs that are "required" so they must available for the PGE to be run. It also has one input that is "alternate." The alternate input can be one of three choices, the first choice is the **primary**, then there are second and third choices.

After the pair of required inputs has become available, the alternate inputs are evaluated as follows:

- If the primary alternate is available, it is used as input and the PGE is scheduled for execution.
- There is a one-hour timer on the primary alternate. If the primary alternate is unavailable, the PGE waits until the primary alternate becomes available or the one-hour timer expires, whichever occurs first.
- If the second alternate is available after the timer for the primary alternate has expired, the second alternate is used as input and the PGE is scheduled for execution.
- There is a four-hour timer on the second alternate. If the second alternate is unavailable, the PGE waits until either the primary alternate or the secondary alternate becomes available or the four-hour timer expires, whichever occurs first.
- If the third alternate is available after the timer for the second alternate has expired, the third alternate is used and the PGE is scheduled for execution.
- There is no timer on the third alternate. If the third alternate is not available, the PGE waits until either the primary alternate, the secondary alternate, or the third alternate becomes available, whichever occurs first.
- The PGE will not start processing until one of the alternates becomes available.
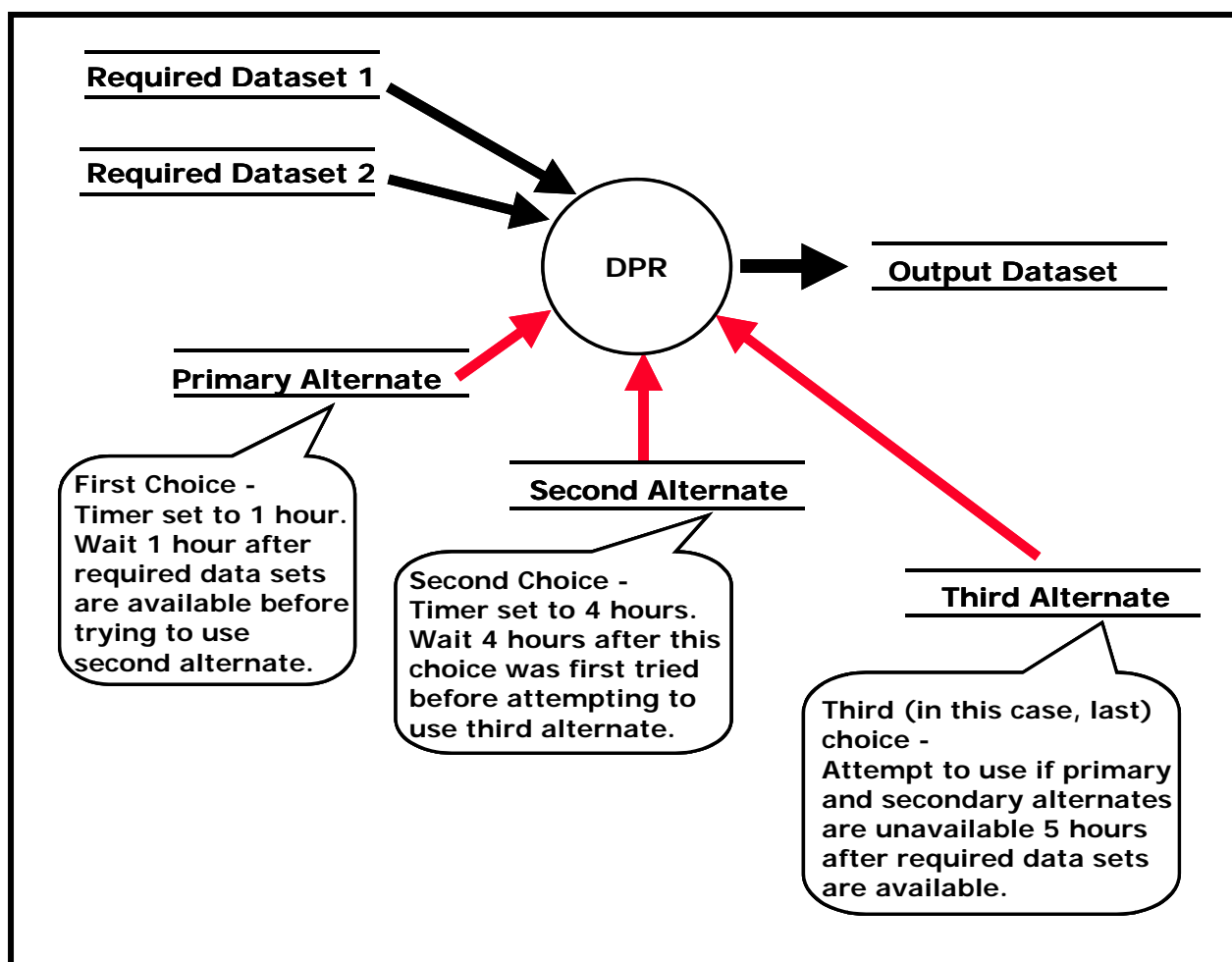
Figure 11.2.3-1. Example of the Alternate Input Production Rule

If instead of an alternate the third input for the PGE had been defined as an optional input, the preceding scenario would have been the same, except that if neither the primary alternate, the second alternate nor the third option was available after the timers had expired, the PGE would not wait; it would be scheduled for execution without the third input. It would run with the two required inputs only.

The Alternate Input and Optional Input Production Rules are additions to settings/syntax put into the ODL files for other production rules. Inputs deemed "optional" or "alternate" can be searched for and acquired by other production rules (e.g., Basic Temporal or Metadata Checks/Query). The syntax for the rules used to search for the inputs must be filled out in addition to the syntax required to make the input an alternate or optional input.

11.2.3.1 PGE Science Metadata ODL File Parameters

The following parameter must be set properly in the applicable PGE science metadata ODL file in order to implement the Alternate Input or Optional Input Production Rule:

- INPUT_TYPE.

In addition, one of the following two ODL objects is used within a PCF_ENTRY to define either the Alternate Input Production Rule or the Optional Input Production Rule:

- ALTERNATE_INPUT object.
- OPTIONAL_INPUT object.

INPUT_TYPE is a type of data defined by a PCF_ENTRY object (i.e., between OBJECT = PCF_ENTRY and END_OBJECT = PCF_ENTRY). It can have one of four possible values, only three of which are used to define an alternate or optional input:

- "Required"
  - A required input.
  - The data must be available or the PGE does not execute.
  - It is the "normal" value for the parameter (i.e., INPUT_TYPE = "Required"); consequently, the input is neither an alternate input nor an optional input.
- "Primary"
  - The primary alternate input.
  - The data is the first choice in a list of alternates.
- "Alternate"
  - An alternate input (except the primary alternate) in a list of alternates.
  - The data is not the first choice in a list of alternates; it is a subsequent choice if the primary (or a higher-priority alternate) is not available.
- "Optional"
  - An optional input.
  - Availability of the data will be checked and if a timer has been specified, execution of the PGE will wait.
  - The PGE can be executed without the data if it is not available.

Although the Alternate Input and Optional Input Production Rules are similar, there are two different ODL objects used to define them within a PCF_ENTRY; i.e., the ALTERNATE_INPUT object and the OPTIONAL_INPUT object.

The ALTERNATE_INPUT object has the following syntax:

```
OBJECT = PCF_ENTRY
   .
   .
   .
 OBJECT = ALTERNATE_INPUT
    .
    .
    .
```

```
        END_OBJECT = ALTERNATE_INPUT
      END_OBJECT = PCF_ENTRY
```

The ALTERNATE_INPUT ODL object surrounds an Alternate Input definition. An OBJECT/END_OBJECT pair separates the parameters defining the Alternate Input from the rest of the parameters defining the PCF_ENTRY. The following parameters define an ALTERNATE_INPUT object:

- CLASS.
- CATEGORY.
- ORDER.
- RUNTIME_PARM_ID.
- TIMER.
- WAITFOR.
- TEMPORAL [not implemented].

CLASS is a simple counter used to differentiate the different ALTERNATE_INPUT objects within the file. Since each ALTERNATE_INPUT object resides within a different PCF_ENTRY object, the CLASS for an ALTERNATE_INPUT object can always be 1.

CATEGORY is the name of the list of alternates to which the ALTERNATE_INPUT belongs. The PDPS uses CATEGORY to associate different alternates within a list. CATEGORY can be set to any string value of 20 characters or less (e.g., CATEGORY = "Snow Ice"). Alternates that are part of the same list should have matching CATEGORY values.

ORDER is the numerical place that the particular alternate holds in the list of alternates. The first choice or Primary Alternate (with the INPUT_TYPE = "Primary") should have ORDER = 1.

RUNTIME_PARM_ID specifies the Logical ID (in the PCF) for which the PGE will find the Logical ID of the alternate chosen. Since all alternates must be contained within different PCF_ENTRY objects, they all must have different Logical IDs (but all alternates within the same CATEGORY should have the same value of RUNTIME_PARM_ID). The RUNTIME_PARM_ID parameter specifies the Logical ID of a runtime parameter that the PGE may read to find out which alternate was chosen for the particular execution of the PGE.

The TIMER parameter specifies how long to wait for the particular alternate before checking for the next alternate in the list. The parameter value is expressed in the format "<Period Type>=<Length of Period>". Note that "Length of Period" can be specified as a positive integer only.

The Alternate Input Production Rule accepts the following "Period Type" values:

- "WEEKS"
  – PDPS should wait for some number of weeks before searching for the next alternate in the list.
  – For example, TIMER = "WEEKS=2" would make PDPS wait two weeks before checking for the next alternate input.

- "DAYS"
  - PDPS should wait for some number of days before searching for the next alternate in the list.
  - For example, TIMER = "DAYS=5" would make PDPS wait five days before checking for the next alternate input.
- "HOURS"
  - PDPS should wait for some number of hours before searching for the next alternate in the list.
  - For example, TIMER ="HOURS=4" would make PDPS wait four hours before checking for the next alternate input.
- "MINS"
  - PDPS should wait for some number of minutes before searching for the next alternate in the list.
  - For example, TIMER = "MINS=5" would make PDPS wait five minutes before checking for the next alternate input.
- "SECS"
  - PDPS should wait for some number of seconds before searching for the next alternate in the list.
  - For example, TIMER = "SECS=2" would make PDPS wait two seconds before checking for the next alternate input.

The WAITFOR parameter specifies whether or not the PGE can be run without the alternate input. Setting WAITFOR = "N" means that the PGE can run without the input if it cannot be found. In a list of alternate inputs, this would have meaning for the last choice only. If WAITFOR = "Y", the PGE is not executed (even after the last alternate timer expires) until one of the alternates in the list can be found.

The TEMPORAL parameter is an unimplemented feature that would allow for searching for alternates from the same time period but a different date. It is currently stored in the PDPS database but is not used.

The OPTIONAL_INPUT object has the following syntax:

```
OBJECT = PCF_ENTRY
   .
   .
   .
   .
   .
   OBJECT = OPTIONAL_INPUT
     .
     .
     .
   END_OBJECT = OPTIONAL_INPUT
END_OBJECT = PCF_ENTRY
```

The OPTIONAL_INPUT ODL object surrounds an Optional Input definition. An OBJECT/END_OBJECT pair separates the parameters defining the Optional Input from the rest of the parameters defining the PCF_ENTRY. The following parameters define an OPTIONAL_INPUT object:

- CLASS.
- CATEGORY.
- ORDER.
- RUNTIME_PARM_ID.
- TIMER.
- TEMPORAL [not implemented].

The parameters that apply to the Optional Input Production Rule are defined in the same way that the corresponding parameters are defined for the Alternate Input Production Rule. However, note that the Optional Input Production Rule has no WAITFOR parameter. It is irrelevant; in fact, the very essence of the Optional Input Production Rule depends on not "waiting for" the last option but going ahead with the execution of the PGE without the unavailable optional input(s).

11.2.4 Minimum/Maximum Number of Granules Production Rule

The Minimum/Maximum Number of Granules Production Rule makes it possible to specify a range of possible granules for a given input or output for a PGE.

- Inputs.
  - Minimum number of granules the PGE needs for full data coverage.
  - Maximum number of granules for the time period.
- Outputs.
  - Minimum number of outputs that the PGE is expected to produce.
  - Maximum number of outputs that the PGE is expected to produce.

For example, a PGE processes data for every 90-minute interval, has a period of 90 minutes, and takes as input a granule with a period of two hours.

- In many instances one granule of the input will satisfy the PGE.
- In other instances, because of the way the two-hour and 90-minute periods overlap, the PGE needs two input granules to cover the time period.
- Therefore:
  - Minimum Number of Granules = 1.
  - Maximum Number of Granules = 2.

The Minimum/Maximum Number of Granules Production Rule is different from most production rules because it works for both input and output granules. It allows the PGE to request of a range of inputs (i.e., 1-10 granules), so that it runs with as few as one granule but with as many as ten granules. If a PGE needs at least three granules of a particular input, the minimum number of granules is defined as three and the PGE is not executed until at least three granules are available.

Optional outputs are defined when the Minimum Number of Granules is set to zero. In such cases the PGE can produce none of the particular type of output and still be considered to have executed successfully. If a PGE has a non-zero value for a Minimum Number of Granules associated with an output, and fails to produce any granules of that output type, it is marked as failed.

Figure 11.2.4-1 provides an illustration of the Minimum/Maximum Number of Granules Production Rule. In the example the PGE processes data related to a one-hour period and takes in both Input 1 and Input 2. Since Input 1 has a PERIOD of 1/2 hour, every PGE run requires two Input 1 granules. Input 2 has a PERIOD of 15 minutes, so there are four Input 2 granules for every PGE run.

The PGE produces three Output 1 granules for each run. In this case it does not produce any Output 2 granules.

Minimum and maximum values can affect each input and output as follows:
- Input 1:
  - If Minimum Granules is set to anything equal to or less than two for Input 1, the PGE is scheduled and executed.
  - If Minimum Granules is set to three, the PGE is not scheduled because there are not enough Input 1 granules to make the minimum.
  - If Maximum Granules is set to anything equal to or greater than two for Input 1, the PGE is scheduled and executed.
  - If Maximum Granules is set to one, the PGE is not scheduled because there are too many Input 1 granules (the number exceeds the maximum that the PGE can process).
- Input 2:
  - If Minimum Granules is set to anything equal to or less than four for Input 2, the PGE is scheduled and executed. If Minimum Granules is set to five, the PGE is not scheduled because there are not enough Input 2 granules to make the minimum.
  - If Maximum Granules is set to anything equal to or greater than four for Input 2, the PGE is scheduled and executed.
  - If Maximum Granules is set to three, the PGE is not scheduled because there are too many Input 2 granules (the number exceeds the maximum that the PGE can process).
- Output 1:
  - If Minimum Granules is set to anything equal to or less than three for Output 1, the PGE is scheduled and executes successfully.
  - If Minimum Granules is set to four, the PGE is marked as failed because it did not produce the expected number of output granules.
  - If Maximum Granules is set to anything equal to or greater than three for Output 1, the PGE is scheduled and executes successfully.
  - If Maximum Granules is set to two, the PGE is marked as failed because it produced too many output granules.
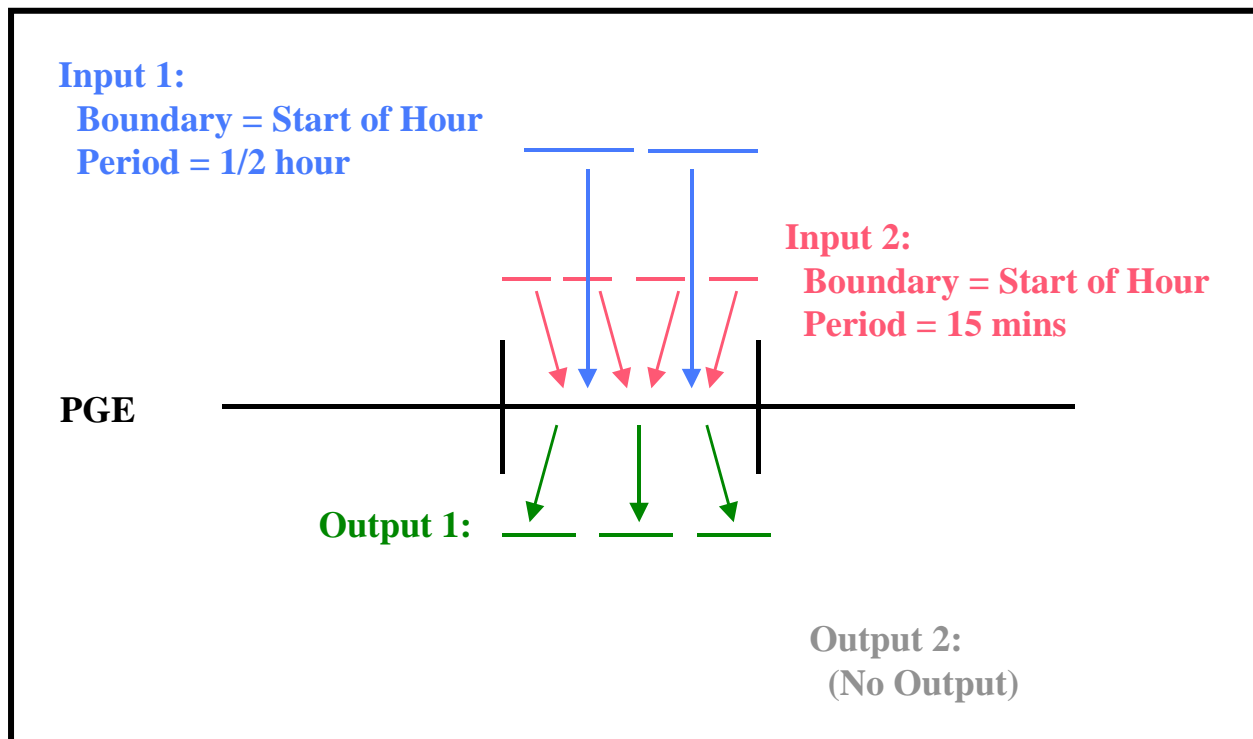
Figure 11.2.4-1. Example of the Minimum/Maximum Number of Granules Production Rule

- Output 2:
    - If Minimum Granules is set to anything other than zero, the PGE is marked as failed because it did not produce the expected number of output granules.
    - If Maximum Granules is set to anything equal to or greater than zero for Output 2, the PGE is scheduled and executes successfully.

The Minimum/Maximum Granules Production Rules are additions to settings/syntax put into the ODL files for other production rules. All Production Rules have a Minimum and Maximum Granule setting for both inputs and outputs, even though both values may be set to one (1).

11.2.4.1 PGE Science Metadata ODL File Parameters

The PGE science metadata ODL file syntax for implementing the Minimum/Maximum Production Rule for **input** data includes the following types of entries:

```
OBJECT = PCF_ENTRY
  .
 PCF_FILE_TYPE =
  .
  .
 MIN_GRANULES_REQUIRED =
 MAX_GRANULES_REQUIRED =
```

.
.
.
                END_OBJECT = PCF_ENTRY

Accordingly, the following parameters must be set properly in order to implement the Minimum/Maximum Production Rule:

- PCF_FILE_TYPE.
- MIN_GRANULES_REQUIRED.
- MAX_GRANULES_REQUIRED.

The PCF_FILE_TYPE parameter is defined by integers in the range of 1 to 8 (inclusive). The integers are codes for the following types of files:

- 1 - product input files.
- 2 - product output files.
- 3 - support input files.
- 4 - support output files.
- 5 - user defined runtime parameters.
- 6 - interim/intermediate input files.
- 7 - interim/intermediate output files.
- 8 - temporary input/output.

For inputs (any PCF_ENTRY with a PCF_FILE_TYPE equal to 1, 3 or 6) the following pair of values must be set for each PCF_ENTRY:

- MIN_GRANULES_REQUIRED
    - Minimum number of granules required for the input.
    - A value of zero (MIN_GRANULES_REQUIRED = 0) would mean that the PGE could execute if no granules for that particular input could be found (in effect, the input is an **optional input**).
    - A value of three (for example) would mean that the PGE must have at least three granules of the input before the PGE can be executed.
- MAX_GRANULES_REQUIRED
    - Maximum number of granules for the input that the PGE is able to successfully process.
    - A value of four (for example) would mean that the PGE would process at most four granules for the input.
    - If MAX_GRANULES_REQUIRED = 4 and more than four granules are found for the given input, the PGE is not executed.

The PGE science metadata ODL file syntax for implementing the Minimum/Maximum Production Rule for **output** data includes the following types of entries:

        OBJECT = PCF_ENTRY
         .
         PCF_FILE_TYPE =
         .

```
              .
        MIN_GRANULE_YIELD =
        MAX_GRANULE_YIELD =
              .
              .
              .
        END_OBJECT = PCF_ENTRY
```

For outputs (any PCF_ENTRY with a PCF_FILE_TYPE equal to 2, 4 or 7) the following pair of values must be set for each PCF_ENTRY.

- MIN_GRANULE_YIELD
  - Minimum number of granules that the PGE produces for the output.
  - A value of zero (MIN_GRANULE_YIELD = 0) means that the PGE produces no granules for the output (the output is an optional output).
  - A value of three (for example) means that the PGE produces at least three granules of the output during a successful execution.
- MAX_GRANULE_YIELD
  - Maximum number of granules that the PGE produces for this output.
  - A value of four (for example) means that at most the PGE produces four granules for the output.
  - Note that sizing of disk space is based on this number, so making it too small could cause problems on the science processor disks.

11.2.5 Optional DPRs Production Rule

The Optional DPRs Production Rule (also called the Data-Scheduled Production Rule) makes the execution of a PGE subject to the availability of a **key input**. The system generates DPRs for every possible instance of the key input data but executes only the DPRs for which data are either produced in data processing or can be acquired from the archive.

The Optional DPRs Production Rule applies to PGEs that process certain kinds of **non-routine data**.

- **Routine Data**
  - Data that can be predicted, that come in at specific intervals and are always of a specified length.
  - Routine data makes it possible for the Basic Temporal Production Rule to schedule PGEs based on their input data.
- **Non-Routine Data**
  - Data that cannot be predicted because they come in at random periods and/or their length is variable.
  - Examples include an "optional" output of an upstream PGE, or data that are archived at random periods (e.g., some forms of ASTER data).

An Optional DPR has as its **key input** a non-routine data type. There are two sets of circumstances that lead to the scheduling of Optional DPRs:

- Every possible time that the input is produced in data processing (i.e., the key input is produced as an "optional" output by an upstream PGE).
- Whenever a new granule (of a particular data type) can be acquired from the archive (e.g., archived data that were inserted at unpredictable times).

NOTE: Key input applies to the Spatial Query Production Rule as well as the Optional DPRs Production Rule. Consequently, it is possible for the Optional DPRs and Spatial Query Production Rules to be used in combination only when they use the same key input.

An example of the first condition starts with a MODIS PGE that produces a certain product only when the input data were collected during the satellite's "Day" mode. A second MODIS PGE is scheduled to use the optional ("Day"-mode) product from the first MODIS PGE as its key input. The second MODIS PGE is scheduled to run after every instance of the first MODIS PGE; however, only the DPRs that can use the optional products resulting from runs of the first MODIS PGE are executed. The remaining DPRs cannot be executed because there is no input data for them.

The second condition is illustrated by ASTER routine processing, which makes use of the Optional DPRs Production Rule to schedule and execute ASTER PGEs for new data that have been archived. (Note that the DAAC ingests and archives ASTER production data from tapes supplied by the ASTER Ground Data System on a frequent but not entirely predictable basis.) When the Production Planner creates a Production Request for an ASTER PGE, the **insertion time** range (i.e., the time period when the desired data were archived) is used as opposed to the **collection time** (when the satellite instrument gathered the data). DPRs specifying the ASTER PGE are scheduled and executed for the data granules that were actually inserted in the archive during the time period specified in the Production Request.

An illustration of the Optional DPRs production rule is presented in Figure 11.2.5-1. In the figure there are two DPRs (i.e., DPR-1 and DPR-2) for the upstream PGE and two DPRs (i.e., OPT-1 and OPT-2) for the PGE subject to the Optional DPRs Production Rule. The "Optional DPRs" PGE takes as input the optional output of the upstream PGE. When it is executed, DPR-1 produces the optional output, so the dependent DPR (OPT-1) is executed. However, OPT-2 is not executed because DPR-2 (on which OPT-2 depends) does not produce the optional output.

The Optional DPRs Production Rule is set up during the SSI&T process. It uses many of the same parameter settings as the Basic Temporal Production Rule so the values specified in the Basic Temporal Production Rule (or other production rules) are set first, then the Optional DPRs Production Rule syntax is added.

Upstream PGE

Optional DPRs

Product One

DPR-1

Optional Product

OPT-1

Product One

DPR-2

Optional Product (not produced)

OPT-2
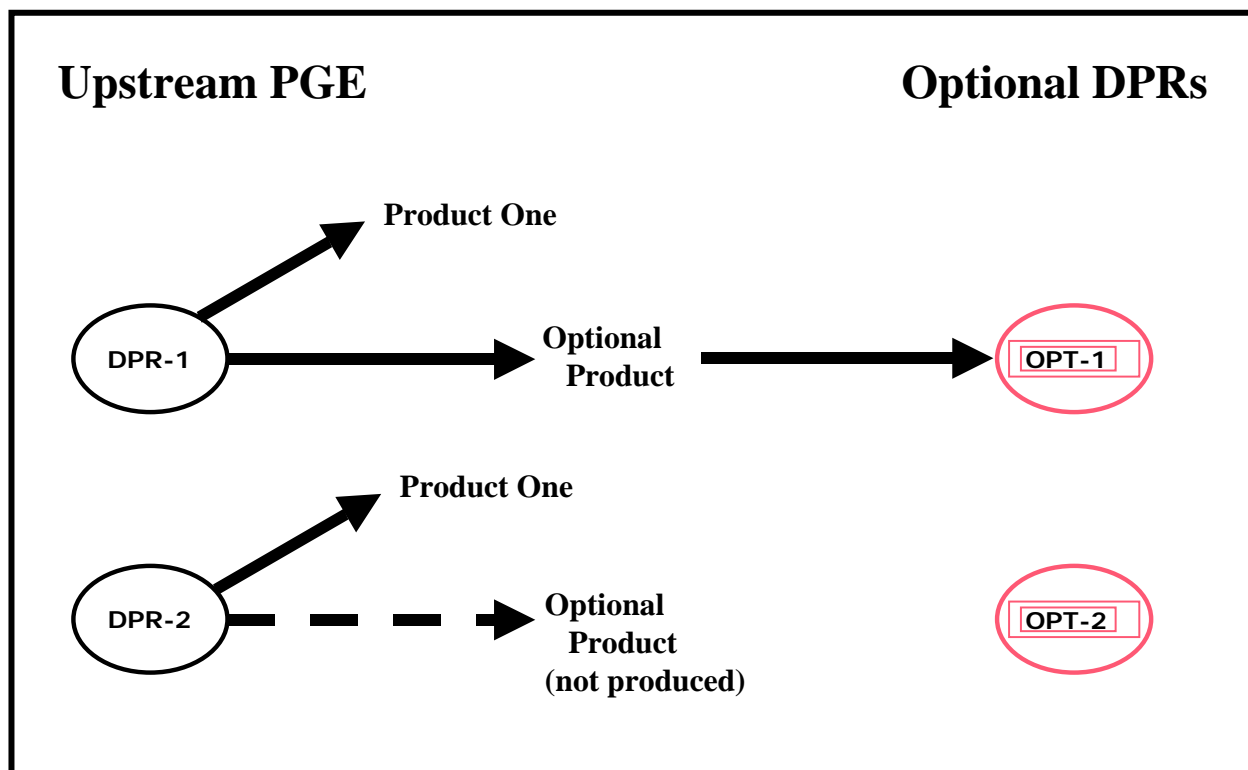
Figure 11.2.5-1. Example of the Optional DPRs Production Rule

11.2.5.1 PGE Science Metadata ODL File Parameters

The following two types of PGE science metadata ODL file entries must be made in order to set up the Optional DPRs Production Rule:

- SCHEDULE_TYPE.
- KEY_INPUT.

The SCHEDULE_TYPE parameter is set as follows:

- SCHEDULE_TYPE = "Data"
  - This demonstrates the appropriateness of the term "Data-Scheduled Production Rule."
  - Other schedule types include Time, Orbit, and Snapshot.

The key input is designated by including the following parameter in the PCF_ENTRY for whichever input is to be the key input:

- KEY_INPUT = "Y"
  - Assigning a value of "Y" to the KEY_INPUT parameter identifies the data as a key input and it is subsequently treated as such.
  - Either assigning a value of "N" to the KEY_INPUT parameter or leaving out the parameter entirely identifies the non-key input data.

– Only one key input is allowed per PGE profile.

The Production Planner's role in the implementation of the Optional DPRs Production Rule was described in the MODIS and ASTER examples previously described and varies with the kind of key input:

- Optional output of an upstream PGE (MODIS example).
  – Production Planner creates Production Requests for the PGE subject to the Optional DPRs Production Rule and specifies the same date/time range as for the upstream PGE.
  – Some of the DPRs generated as a result of the Production Request will never run due to lack of input data.
- Ingested on an irregular time schedule (ASTER example).
  – Production Planner specifies the data insertion time range when creating Production Requests.
  – All DPRs generated as a result of the Production Requests should be capable of running.

11.2.6 Metadata Checks and Metadata Query Production Rules

The Metadata Checks and Metadata Query Production Rules are similar in definition and use. Both production rules allow the PGE to specify granule-level metadata values that define whether the PGE can accept one (or more) of its inputs. The rules differ only in the results of metadata search performed.

- Metadata Checks Production Rule.
  – When PLS requests the Science Data Server to search for the input(s), the Science Data Server "checks" the metadata of all granules that match the time frame with respect to the value(s) allowed by the PGE.
  – If any granule fails to match the specified value(s), the PGE is not executed.
- Metadata Query Production Rule.
  – When PLS requests the Science Data Server to search for the input(s), the Science Data Server adds to the query the metadata value(s) desired by the PGE.
  – Only the granules that match the time frame of the PGE plus the granule-level metadata value(s) specified by the PGE are staged for the PGE to use as input.
  – If no granules are found matching the conditions and the input is not optional, the PGE is not executed.
- Example of Metadata Checks:
  – A MODIS PGE is run when the Percent Cloud Cover of its inputs is greater than 25 percent.
  – The Metadata Checks Production Rule is used to specify the granule-level metadata value of greater than 25.
  – When the PGE is scheduled and is ready to start, two granules match the timeframe of the Production Request for the input with the Metadata Check.
  – If both granules have a Percent Cloud Cover greater than 25 percent, execution of the PGE starts and both granules are staged.

- If one of the granules has a Percent Cloud Cover of 15 percent, the PGE is not executed.
- Example of Metadata Query:
    - A MODIS PGE is run when as many granules as possible of one of its inputs have a QA Value = "Good".
    - The Metadata Query Production Rule is used to specify the granule-level metadata value = "Good".
    - When the PGE is scheduled and is ready to start, two granules match the time frame of the production request for the input with the Metadata Query.
    - If both granules have a QA Value = "Good", execution of the PGE starts and both granules are staged.
    - If one of the granules has a QA Value = "Bad", the PGE executes but with only one granule (the one with QA Value = "Good").

The Metadata Checks and Metadata Query Production Rules are used in conjunction with the times specified in the Basic Temporal Production Rule or other production rules. The Metadata Check or Query is added information that further refines what granules are sought by the PGE.

The Metadata Checks and Metadata Query Production Rules work differently depending on whether the Production Request is created for a time in the past or for a time in the future.

- Past data.
    - The Production Request Editor performs the Metadata Query or Metadata Check immediately.
    - Depending on what other production rules may be applicable, if no data is found to match the "query" (or data is found that does **not** match the "check"), the DPR fails to be created.
- Future data.
    - The Metadata Query/Metadata Check is put off until the DPR Data Collection Stop Time passes **plus** the value defined in the ODL for QUERY_DELAY.
    - The delay allows the Metadata Query/Metadata Check to be put off until it's likely that all matching data will be present.

Multi-Granule ESDTs are a special case of the Metadata Query Production Rule. Multi-Granule ESDTs are used for PGE inputs or outputs when more than one granule of the same ESDT exists for the same temporal range (time period). The Multi-Granule ESDT mechanism employs a metadata parameter to differentiate between the "equal in time" granules. A metadata parameter is selected that is unique across granules for the same time period and that is used by PDPS to keep track of which granule is which when the granules are produced. Later, if only one pair of granules for a particular time period is needed as input to the PGE, the Metadata Query is used to ensure that PDPS schedules the correct granule as input.

Data Day is actually an addition to the Metadata Query Production Rule involving **runtime parameter** values. Two settings (i.e., "Start Data Day" and "End Data Day") allow a PGE to perform a Metadata Query for the start of the Data Day and the end of the Data Day. The Start Data Day and End Data Day are calculated by subtracting twelve (12) hours from the starting day for which the PGE is executing and adding twelve (12) hours onto the ending day for which

the PGE is running. So if the PGE is running from 00:00:00 on 07/04 to 00:00:00 07/05 then the Start Data Day = 07/03 12:00:00 and the End Data Day = 07/06 12:00:00.

Using runtime parameter values is a capability of the Metadata Query and Metadata Checks Production Rules. Rather than use a hard-coded value for the check or query, a value computed from one of the other production rules can be used.

Figure 11.2.6-1 illustrates the Metadata Checks and Metadata Query Production Rules. If no Metadata Check or Query were applicable, the PGE shown in the figure would use three granules of input (i.e., Granules A through C). However, let us assume that the metadata value to be checked/queried is %CloudCover. Each granule has a different value for %CloudCover.
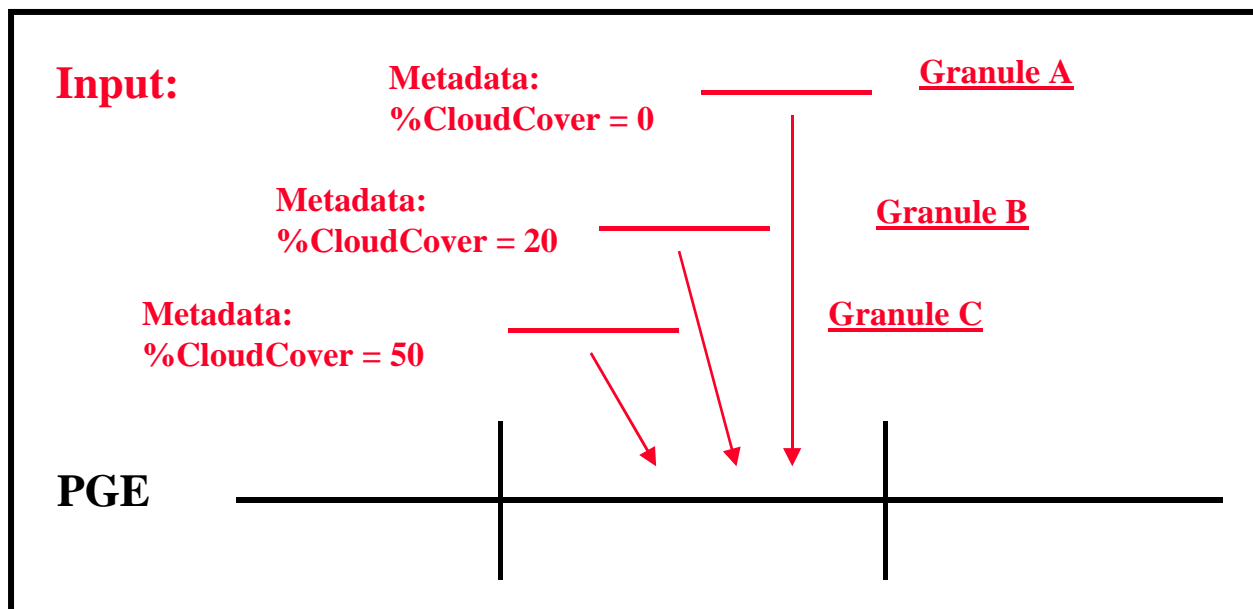
Figure 11.2.6-1. Example of the Metadata Checks and Query Production Rules

The following results demonstrate the differences between the Metadata Checks and Metadata Query Production Rules, especially with respect to the number of inputs that the PGE receives when different values are specified:

- Metadata Check of %CloudCover < 80:
  - In this case all three granules are acquired and the PGE is scheduled and executed.
- Metadata Query of %CloudCover < 80:
  - All three granules are acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 50:
  - The PGE is not scheduled because only one of the three granules (Granule C) meets the criterion.

611-EMD-001, Rev. 02

- Metadata Query of %CloudCover = 50:
  - Granule C is found and if the PGE's Min/Max Granules parameters are set to allow one granule, that one granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover < 50:
  - The PGE is not scheduled because only two of the three granules (Granule A and B) meet the criterion.
- Metadata Query of %CloudCover < 50:
  - Granules A and B are found and if the PGE's Min/Max Granules parameters are set to allow two granules, the granules are acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover <= 50:
  - The PGE is scheduled and executed because all three granules meet the criterion.
- Metadata Query of %CloudCover <= 50:
  - All three granules are found and acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 20:
  - The PGE is not scheduled because only one of the three granules (Granule B) meets the criterion.
- Metadata Query of %CloudCover = 20:
  - Granule B is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover < 20:
  - The PGE is not scheduled because only one of the three granules (Granule A) meets the criterion.
- Metadata Query of %CloudCover < 20:
  - Granule A is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 10:
  - The PGE is not scheduled because none of the three granules meets the criterion.
- Metadata Query of %CloudCover = 10:
  - The PGE is not scheduled because no granules are returned from the query (unless Minimum Granules is set to 0).

Note that there can be more than one Metadata Check or Metadata Query on a given input. In the preceding example, a Metadata Check on %CloudCover can be combined with a Metadata Query on another parameter to further limit the input.

The Metadata Checks and Metadata Query Production Rules are additions to settings/syntax put into the ODL files for other production rules. The addition of a Metadata Check or a Metadata Query to an input means that other production rules used to evaluate that input will be applied in combination with the Metadata Check or Metadata Query.

11.2.6.1 PGE Science Metadata ODL File Parameters

At the main ODL level (not inside any objects) both the Metadata Checks and Metadata Query Production Rules use the following parameter to determine when to perform the check/query:

- QUERY_DELAY

The parameter is a positive integer (number of seconds) that defines the amount of time to wait before performing the Metadata Query or Metadata Check. When planning DPRs to run on future data, the Production Request Editor sets a timer for all Metadata Checks/Queries at the Data Collection Stop Time (end of the DPR) **plus** the value specified for the QUERY_DELAY parameter. The value adds time to the Query/Check so the PGE can be assured that all matching data will be present when the Query/Check is performed.

Both production rules use also need the following setting in the PCF_ENTRY to ensure that Planning and Data Processing consider metadata values when acquiring input data:

```
OBJECT = PCF_ENTRY
 .
 .
 QUERY_TYPE = "Metadata"
 .
END_OBJECT = PCF_ENTRY
```

QUERY_TYPE tells Planning and Data Processing how to retrieve the data for the input. For inputs based on metadata (whether Metadata Check or Metadata Query), the value is set to "Metadata".

Although the Metadata Checks and Metadata Query Production Rules are similar, there are two different ODL objects used to define them within a PCF_ENTRY in the PGE science metadata ODL file; i.e., the METADATA_CHECKS object and the METADATA_QUERY object.

The METADATA_CHECKS object has the following syntax:

```
OBJECT = PCF_ENTRY
 .
 .
 .
 .
 .
 OBJECT = METADATA_CHECKS
  .
  .
  .
 END_OBJECT = METADATA_CHECKS
END_OBJECT = PCF_ENTRY
```

The METADATA_QUERY object has the same syntax except "METADATA_QUERY" replaces "METADATA_CHECKS" in every instance.

Most of the following parameters must be set in the PGE science metadata ODL file within the METADATA_CHECKS or METADATA_QUERY ODL object (as applicable) in order to implement either the Metadata Checks or Metadata Query Production Rule:

- CLASS.
- PARM_NAME.
- OPERATOR.
- VALUE.
- DATABASE_QUERY.
- KEY_PARAMETER_NAME (optional).
- KEY_PARAMETER_VALUE (optional).

CLASS is a simple counter used to differentiate the different Metadata Checks or Metadata Query objects within the file. Since each Metadata Checks or Metadata Query object resides within a different PCF_ENTRY object, the CLASS for an METADATA_CHECKS or METADATA_QUERY object can always be 1 (e.g., CLASS = 1).

PARM_NAME is the name of the metadata parameter on which the check or query is to be performed. The value specified for PARM_NAME (e.g., PARM_NAME = "%CloudCover") must be part of the granule-level metadata of the ESDT. In addition, it must match the parameter name specified in the ESDT science metadata ODL file.

OPERATOR is the operator (e.g., OPERATOR = "==") on which the check/query is to be performed. The following values are valid for OPERATOR:

- ">"
  - Value in metadata must be greater than.
- "<"
  - Value in metadata must be less than.
- ">="
  - Value in metadata must be greater than or equal to.
- "<="
  - Value in metadata must be less than or equal to.
- "=="
  - Value in metadata must be equal to.
- "!="
  - Value in metadata must be **not** equal to.

VALUE is the value (e.g., VALUE = 50) against which the metadata parameter (defined by PARM_NAME) is compared (using the operator specified by the OPERATOR parameter). The value for the VALUE parameter should be the type of data (e.g., integer or string) as defined in the ESDT ODL metadata for the parameter.

DATABASE_QUERY indicates whether the value for the Metadata Check or Query should be retrieved from the PDPS database rather than through the use of the VALUE parameter. Specifying DATABASE_QUERY permits **runtime parameter values** to be used for Metadata Query or Metadata Checks. The following values are valid for the DATABASE_QUERY parameter:

- "NONE"
    - Use the value in the VALUE parameter; no value from the PDPS database is used.
- "PATH NUMBER"
    - Use the Path Number (0-233) of the orbit for which the PGE is scheduled.
- "ORBIT NUMBER"
    - Use the Orbit Number of the orbit for which the PGE is scheduled.
- "TILE ID"
    - Use the Tile ID of the current Data Processing Request.

KEY_PARAMETER_NAME is an optional parameter that is used to specify the container within a multi-container metadata group (i.e., the MeasuredParameters metadata group in most ESDTs). The KEY_PARAMETER_NAME (e.g., KEY_PARAMETER_NAME = "ParameterName" for metadata checks or queries within the MeasuredParameters group) in conjunction with the KEY_PARAMETER_VALUE allows PDPS to determine which container within the multi-container group is to be the object of the check or query. KEY_PARAMETER_NAME is **not** used for product-specific attributes.

KEY_PARAMETER_VALUE is an optional parameter that is used to specify the **value** (e.g., KEY_PARAMETER_VALUE = "LandCoverage") for the container within a multi-container metadata group (i.e. the MeasuredParameters metadata group in most ESDTs). The

KEY_PARAMETER_VALUE in both the PGE science metadata ODL file and ESDT science metadata ODL file must match.

For Multi-Granule ESDT inputs the METADATA_QUERY object (consisting of entries for CLASS, PARM_NAME, OPERATOR, VALUE, DATABASE_QUERY, KEY_PARAMETER_NAME, and KEY_PARAMETER_VALUE) is replaced by the following parameter under the PCF_ENTRY object in the PGE science metadata ODL file:

- DISTINCT_VALUE.

The DISTINCT_VALUE must be set to the value of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. In addition, the input or output defined by the PCF_ENTRY must have a corresponding DISTINCT_PARAMETER entry in the appropriate ESDT science metadata ODL file.

11.2.6.2 ESDT Science Metadata ODL File Parameters

The METADATA_DEFINITION ODL object surrounds the definition for Metadata Checks or Metadata Query information within the ESDT science metadata ODL file. An OBJECT/END_OBJECT pair is needed to separate the parameters defining the Metadata Definition from the rest of the parameters defining the ESDT with the following syntax:

> OBJECT = METADATA_DEFINITION
> .
> .
> .
> END_OBJECT = METADATA_DEFINITION

A METADATA_DEFINITION object can match multiple Metadata Checks or Metadata Query objects in various PGE science metadata ODL files. There is no difference between the two production rules with respect to the parameters that need to be set in the ESDT science metadata ODL file. Most of the following parameters must be set:

- CLASS.
- PARM_NAME.
- CONTAINER_NAME.
- TYPE.
- KEY_PARAMETER_NAME (optional).
- KEY_PARAMETER_VALUE (optional).

CLASS is a simple counter used to differentiate the different Metadata Definition objects within the file. Each Metadata Definition object within the file must have a **different** CLASS value.

PARM_NAME is the name of the Metadata parameter on which the check or query will be performed. The value specified for PARM_NAME must be part of the granule-level metadata of the ESDT. It must also match the parameter name specified in the PGE science metadata ODL file(s).

CONTAINER_NAME is the name of the Metadata Group within which the metadata parameter defined by PARM_NAME is contained. For product-specific attributes CONTAINER_NAME is set to the string "AdditionalAttibutes" (i.e., CONTAINER_NAME = "AdditionalAttibutes").

TYPE indicates the type of data within the metadata parameter. The following values are valid for TYPE:

- "INT"
  - Integer data.
- "FLOAT"
  - Floating-point data.
- "STR"
  - String or character data.
  - Note that dates and times are considered string data.

KEY_PARAMETER_NAME and KEY_PARAMETER_VALUE are optional parameters that are used to specify the value for the container within a multi-container metadata group (i.e., the MeasuredParameters metadata group in most ESDTs). They allow PDPS to determine which container within the multi-container group is to be the object of the check or query. There must be a match between the KEY_PARAMETER_VALUE in both the ESDT science metadata ODL file and the PGE science metadata ODL file for any PCF_ENTRY that uses the ESDT.

The ESDT science metadata ODL file for an input specifying Multi-Granule ESDTs needs to have the following parameter added:

- DISTINCT_PARAMETER.

The DISTINCT_PARAMETER must be set to the name of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. A corresponding METADATA_DEFINITION must be created to help PDPS find the specified metadata parameter when querying the Science Data Server.

11.2.7 Spatial Query Production Rule

The Spatial Query Production Rule allows a PGE to select input(s) based on the spatial coverage of another input (called the **key input**). The PDPS queries the Science Data Server for the spatial coverage of the key input, then uses it in acquiring any subsequent inputs that the PGE has requested that have the same spatial coverage.

- Example:
  - Level 0 input data for an ASTER DPR covers a small section of the Earth.
  - The PGE requires ancillary data that covers the same area to complete its processing.
  - The PGE uses the Spatial Query Production Rule to mark the geographic input as its key input.
  - The PGE specifies that the ancillary input is to be retrieved for the same spatial coverage as that of the key input.
  - When PDPS finds an input granule for the PGE, it performs a Spatial Query to acquire the ancillary input with the same spatial coverage as that of the key input.

Without specifying coordinates, PDPS can match inputs against the spatial constraint of the key input, and give to a PGE only those granules, which overlap in area.

Spatial Pad is an addition to the Spatial Query Production Rule. Spatial Pad is a means of padding the spatial constraints of the key input. The specified pad is added to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.

Due to the limitations in the current version of the SQS COTS package, the spatial region defined (i.e., the original size of the polygon combined with the size of the pad) must not exceed 60 degrees in latitude and longitude. If no spatial pad is specified or the pad is equal to zero, normal spatial query is used.

The **Spatial Query** Production Rule (unlike most time-base production rules) will work with both static and dynamic granules. Thus the Digital Elevation Models (DEMs) can be retrieved via a Spatial Query.

Figure 11.2.7-1 is an illustration of the Spatial Query Production Rule. The figure shows a PGE that has two input types, one of which is the key input. The other type of input has granules labeled with the names of various colors. One granule (i.e., "green") of the key input is found.
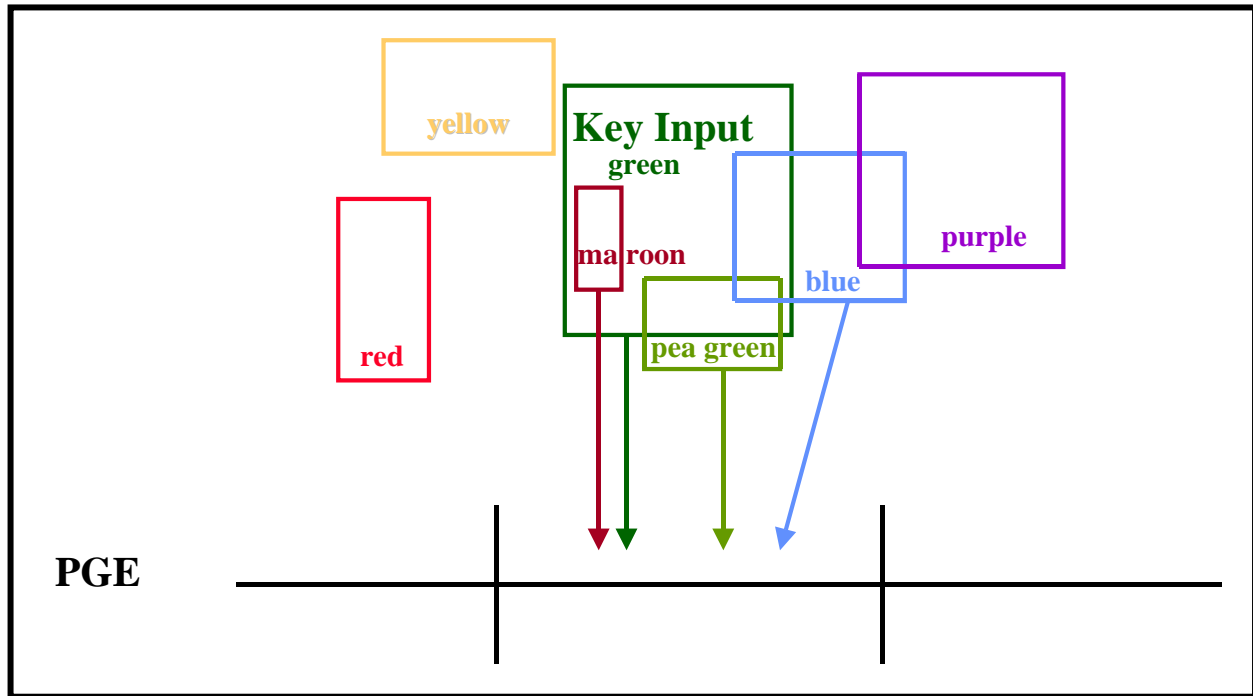
Figure 11.2.7-1. Example of the Spatial Query Production Rule

The spatial coordinates of the granule are retrieved and all inputs of the second ESDT are checked for overlap with the key input's coordinates.

Assuming that all granules relate to the same time period, the granules are evaluated as follows:

- The "yellow" granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input.
- The "red" granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input.
- The "blue" granule is retrieved as an input because its spatial coordinates overlap with those of the key input. Part of its spatial constraint is within the constraint of the key input.
- The "maroon" granule is retrieved as an input because its spatial coordinates overlap with those of the key input. The spatial constraint of this granule is completely within the constraint of the key input.

- The "pea green" granule is retrieved as an input because its spatial coordinates overlap with those of the key input. Part of its spatial constraint overlaps with that of the key input.
- The "purple" granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input. It does not matter that it overlaps with another input that is accepted (i.e., the "blue" granule).

The Spatial Query Production rule is somewhat of an addition to other production rules. As such, it needs the same parameter settings as the Basic Temporal Production Rule. The values specified in the Basic Temporal Production Rule (or other production rules) are set first, and then the Spatial Query Production Rule syntax is added.

11.2.7.1 Spatial Query with Spatial Pad

Figure 11.2.7-2 is an illustration of Spatial Query with Spatial Pad. As in Figure 11.2.7-1 the PGE has two input types, one of which is the key input (in green). The Spatial Pad value (the dotted line in Figure 11.2.7-2) is added around the key input to increase the area covered by the spatial coordinates. Then all inputs of the second ESDT are checked for overlap with the padded coordinates.
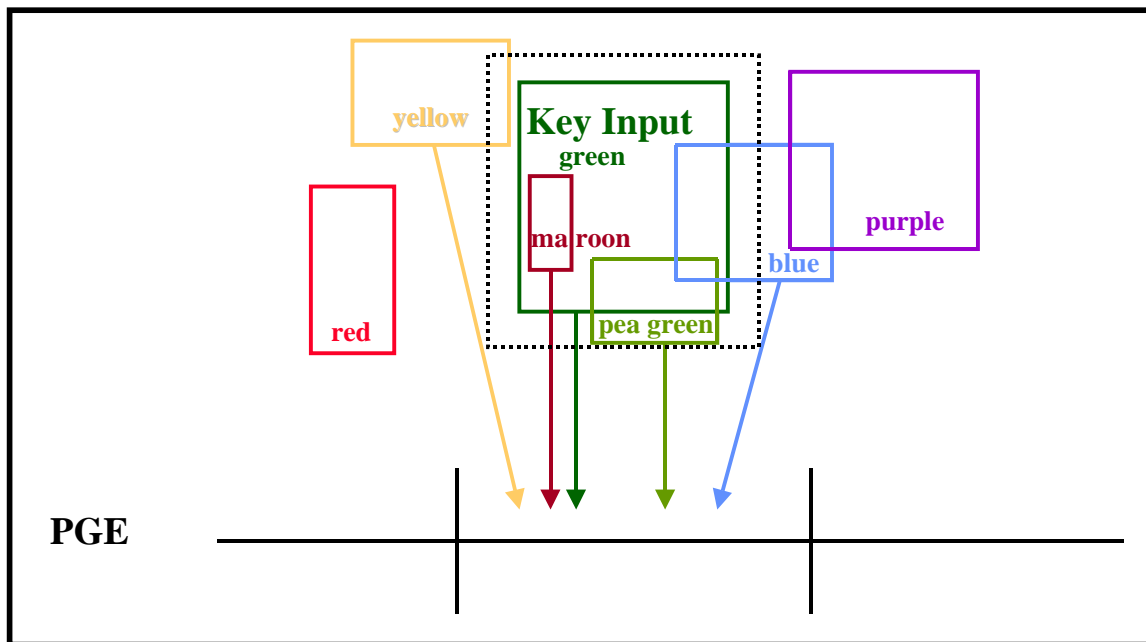
Figure 11.2.7-2. Example of the Spatial Query Production Rule with Spatial Pad

Assuming that all granules relate to the same time period, the granules are evaluated as follows:
- The "yellow" granule is retrieved as an input; although its spatial coordinates do not overlap with those of the key input, they do overlap the padded area (the dotted line).

- The "red" granule is not retrieved as an input because its spatial coordinates do not overlap with either the key input or the pad.
- The "blue" granule is retrieved as an input because its spatial coordinates overlap with the key input. Part of its spatial constraint is within the constraint of the key input.
- The "maroon" granule is retrieved as an input because its spatial coordinates overlap with the key input. The spatial constraint of this granule is completely within the constraint of the key input.
- The "pea green" granule is retrieved as an input because its spatial coordinates overlap with the key input. Part of its spatial constraint overlaps with that of the key input.
- The "purple" granule is not retrieved as an input because its spatial coordinates do not overlap with the key input. It does not matter that it overlaps with another input that is accepted, the "blue" granule.

11.2.7.2 PGE Science Metadata ODL File Parameters

Each input file is defined by a separate PCF_ENTRY object in the PGE science metadata ODL file. To mark a particular input as being subject to a Spatial Query the QUERY_TYPE parameter needs to be set to "Spatial" within that input's PCF_ENTRY:

```
OBJECT = PCF_ENTRY
  .
  .
  .
QUERY_TYPE = "Spatial"
  .
  .
  .
END_OBJECT = PCF_ENTRY
```

QUERY_TYPE indicates what type of query is to be done to acquire the input defined by the PCF_ENTRY object. Valid values are as follows:

- "Temporal" - Input is acquired based on time.
  - The Basic Temporal and/or the Advanced Temporal Production Rules is/are used to get the input.
  - "Temporal" is the value that is assumed if the parameter is left out of the PCF_ENTRY object.
- "Spatial" - Input is acquired based on spatial coordinates (as well as time).
  - An input must be designated the key input to be used in determining the spatial constraints of the search.
  - "Spatial" is the value specified for each input that uses the Spatial Query Production Rule.
- "Tile" - Input is acquired by the spatial definition of a tile.
  - Refer to the Tiling Production Rule for additional information.

- "Already Created Tile" - Input is acquired based on the tile ID of an already created tile.
  - Refer to the Tiling Production Rule for additional information.

The PCF_ENTRY object for the key input includes the following syntax:

```
OBJECT = PCF_ENTRY
.
.
KEY_INPUT = 'Y'
.
.
END_OBJECT = PCF_ENTRY
```

The KEY_INPUT is the input on which the spatial queries for other inputs will be based. When a KEY_INPUT parameter is assigned a value of "Y" the corresponding input is designated a key input and is treated as such. A value of "N" or leaving out the parameter entirely specifies a non-key input. Only one (1) key input is allowed per PGE Profile.

If a pad is designated as an addition to the spatial constraints of the KEY_INPUT, the SPATIAL_PAD parameter must be set. The pad value is added to the key input's spatial constraints. The value for the SPATIAL_PAD parameter is set within the PCF_ENTRY defining the key input using the following syntax:

```
OBJECT = PCF_ENTRY
.
.
KEY_INPUT = 'Y'
SPATIAL_PAD =
.
.
END_OBJECT = PCF_ENTRY
```

The SPATIAL_PAD represents the number of kilometers (from 0.0 to 1000.0) that should be added to the Spatial Query. The value is a real number between 0 and 1000 and if not specified, it is set to 0.0 (and thus no pad is added to the spatial queries).

To allow for a temporal offset when performing a spatial query, the SPATIAL_TIME_DELTA parameter is specified to add or subtract time (in seconds) from the query. Consequently, the temporal range for the query may be different from the temporal range of the key input. The value for the SPATIAL_TIME_DELTA parameter is set within the PCF_ENTRY for the spatial input:

```
OBJECT = PCF_ENTRY
.
.
.
QUERY_TYPE = "Spatial"
```

```
        SPATIAL_TIME_DELTA =
        .
        .
    END_OBJECT = PCF_ENTRY
```

The SPATIAL_TIME_DELTA adds or subtracts seconds to/from the start and stop times of the DPR when performing the spatial query for the input defined by the PCF_ENTRY. In effect it widens the temporal query. For example…

- DPR Time = 2:00:00 to 3:00:00
- SPATIAL_TIME_DELTA = 20 [seconds]
- Query time for spatial input: 1:50:40 to 3:00:20

11.2.8 Closest Granule Production Rule

The Closest Granule Production Rule allows a PGE to request the nearest input granule from the Data Processing Request time. The PDPS requests a search forward or backward for a specified period of time until it finds a granule that matches the request. However, there is a limit to the number of queries that are performed. The number of queries and the period length of the query are specified during SSI&T.

- Example:
    – A PGE processes data at daily intervals and could use a particular type of calibration granule that would allow it to determine the nearest parameters of the instrument.
    – Although most calibration coefficients are defined as static granules, in this case there is a dynamic granule that is received about once a month.
    – The closest such granule would be optimal, so the PGE uses the Closest Granule Production Rule to search forward or backward from the time of the DPR to find the nearest calibration granule.

The Closest Granule Production Rule supersedes the Most Recent Granule Production Rule. The latter allowed the search for inputs to go backward in time from the start of the DPR. The Closest Granule Production Rule allows the search for input granules to go either backward or forward in time, increasing the flexibility of the rule. The Closest Granule Production Rule has all of the ability of Most Recent Granule, plus the ability to search forward in time for input data.

The Closest Granule Production Rule uses three values to determine what to query. The three values are concerned with the period of time to query, the direction of the query, and the number of queries allowed.

- Period (Offset).
    – Tells the PDPS software the query duration.
- Direction.
    – Indicates whether the query goes forward (positive) or backward (negative) in time.

- − In the PlDataTypeReq table in the PDPS database, the Direction and the Period information are combined, so the sign of value (+ or -) indicates the Direction and the magnitude indicates the Period.
- Maximum Number of Queries.
  - − Tells the PDPS software how many time periods (as defined by the Offset) to search (either forward or backward in time) for a matching granule.

The PDPS does a Basic Temporal query before using Closest Granule to find the input. If the desired input is not found within the time period of the DPR, PDPS performs a query (in the specified direction) against the Science Data Server for the period defined by the offset. Again, if no matching granule is found, PDPS repeats the query, going backward or forward in time by the value specified in the offset. If no acceptable granule has been found before the maximum number of queries is reached, PDPS fails to generate the DPR due to insufficient input data. However, there is a special case for the forward search; i.e., when the next search interval exceeds the current time, the search stops at the current time.

If the DPR is planned for a future time, the DPR is created using placeholder granules and a timer is activated. When the current time reaches the stop time of DPR, the timer invokes the Closest Granule method to search for the actual granules, which are then used to replace the dummy granules.

Figure 11.2.8-1 illustrates the Closest Granule Production Rule. In the example, the PGE has a boundary of "start of day" and a period of one hour, so it is scheduled to run for one hour's worth of input data. The input has a period of one hour, and can come in at any hour of the day. Consequently, the PGE requests one granule of input.
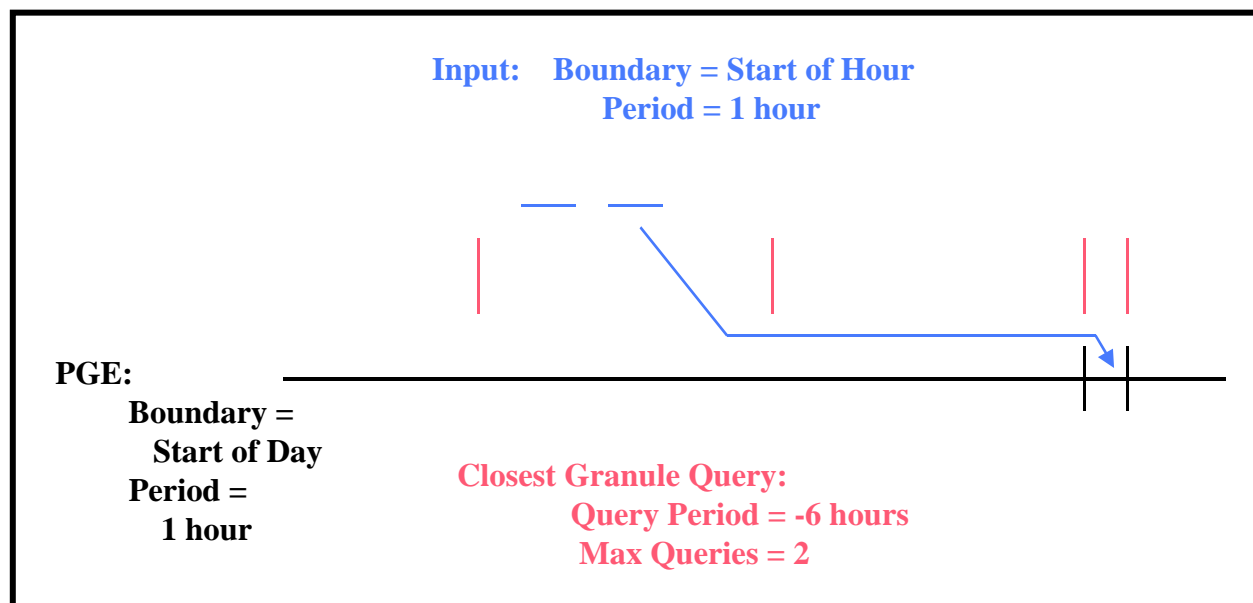
Figure 11.2.8-1. Example of Closest Granule Production Rule

The PGE has defined the Closest Granule Production rule with a 6-hour period and a direction of backward, meaning that it queries back in time in six-hour intervals. The maximum number of queries is two. The PDPS performs a query for the input based on the time period of the DPR. Not finding any matching data, it uses the Closest Granule information to query for a six-hour period beginning six hours before the start time of the DPR. Again nothing is found, so a second Closest Granule query is performed, this one six hours before the last Closest Granule query. The second query results in the discovery of two matching granules. The PDPS selects the granule that is later in time and schedules the PGE to use it as input.

If the Closest Granule Production Rule were used in conjunction with the Minimum/Maximum Number of Granules Production Rule, it might be possible for both granules to be selected in the previously described Closest Granule query. If the example included setting the Maximum Number of Granules to two, both granules would be selected as input to the PGE.

The Closest Granule Production Rule needs the same parameter settings as the Basic Temporal Production Rule. The values needed for the Basic Temporal Production Rule must be set before the Closest Granule Production Rule syntax is added.

11.2.8.1 PGE Science Metadata ODL File Parameters

In addition to the parameter settings for the Basic Temporal Production Rule, the following parameters must be set within the appropriate PCF_ENTRY in the PGE science metadata ODL file in order to implement the Closest Granule Production Rule:

- CLOSEST_QUERY_OFFSET.
- CLOSEST_QUERY_RETRIES.
- CLOSEST_QUERY_DIRECTION.

As previously mentioned, if it is necessary to distinguish among granules with the same ESDT (e.g., among granules associated with different sensors for an instrument), the following parameter must be added to the PCF_ENTRY in the PGE science metadata ODL file:

- DISTINCT_VALUE.

CLOSEST_QUERY_OFFSET is the offset added to or subtracted from the Data Start Time of the DPR and uses as the query for the requested input data type. The specified value has the format "<Period Type>)=<Length of Period>" (e.g., CLOSEST_QUERY_OFFSET = "HOURS=6"). The following "Period Type" values are used in implementing the Closest Granule rule:

- "WEEKS"
  - Offset is some number of weeks.
  - For example, "WEEKS=2" would be a 14-day offset.
- "DAYS"
  - Offset is some number of days.
  - For example, "DAYS=5" would be a 120-hour offset.
- "HOURS"
  - Offset is some number of hours.
  - For example, "HOURS=4" would be a 240-minute offset.

- "MINS"
  - Offset is some number of minutes.
  - For example, "MINS=5" would be a 300-second offset.
- "SECS"
  - Offset is some number of seconds.

CLOSEST_QUERY_RETRIES is the maximum number of Closest Granule queries before the DPR fails due to insufficient input data. The specified value is an integer value that is limited only by the maximum size of an integer on the executing hardware.

CLOSEST_QUERY_DIRECTION is the direction ("Forward" or "Backward") in time to search the CLOSEST_QUERY_OFFSET for a matching granule. The following "Direction" values are used in implementing the Closest Granule rule:

- "Forward"
  - Search forward in time (from the end of the DPR).
- "Backward"
  - Search backward in time (from the start of the DPR).

Note that the longer the offset value or the greater the number of retries, the more time that each query requires due to search time at the Science Data Server and processing time of any granules returned. The combination of a large offset with a large number of retries can (if no data granules are found) consume a lot of time while failing to generate a DPR.

The DISTINCT_VALUE (if used) must be set to the value of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. In addition, the input or output defined by the PCF entry must have a corresponding entry identifying the metadata parameter in the ESDT science metadata ODL file.

11.2.9   Orbital Processing Production Rule

The Orbital Processing Production Rule is similar to the Basic Temporal Production Rule in that both define the time period for the inputs and outputs of the PGE. The difference is that the Orbital Processing Production Rule uses the orbit of the spacecraft to determine that time period. A PGE that processes data related to every orbit of a satellite uses data related to a time period that is computed from the orbit of that satellite.

- Example:
  - A PGE processes Level 0 data related to each orbit of the Terra satellite.
  - The Terra satellite has an orbital period of 98 minutes so the PGE is scheduled to process data for each 98-minute interval.
  - Since Level 0 data are received every two hours, the data staged for the PGE include every Level 0 granule that falls within the 98-minute PGE interval.
  - Only one granule of Level 0 data is relevant to some 98-minute orbits.
  - Two granules of Level 0 data are relevant to other 98-minute orbits.

The Orbital Processing Production Rule uses the "period" and "boundary" concept just like the Basic Temporal Production Rule. The difference is that for Orbital Processing, the orbit of the spacecraft is taken into account when a PGE or its data are marked as **orbit scheduled**.

When responding to a Production Request for orbit-scheduled processing, PDPS determines the orbit of the satellite via information provided during the SSI&T process. The information (stored in the PDPS database) gives the start time and length of a particular orbit or set of orbits. PDPS extrapolates (or interpolates in the case of an orbit between two orbital periods stored in the database) the start and end times of the PGE that is specified in the Production Request. Data are sought on the basis of the derived start and stop times and the appropriate data granule(s) is/are staged before the PGE is executed.

Orbit model is a model of the satellite's orbits that allows PDPS to perform extrapolations for the Orbital Processing Production Rule. The model is a combination of a database table and a simple algorithm. The database table stores Orbit Number/Orbit Start Time/Orbital Period combinations. The algorithm uses the data to compute the same type of data relevant to subsequent orbits. For example, if Orbit Number 1000 is defined in the database table, the algorithm can calculate the start and end times for Orbits 1001, 1015, 2000, etc.

The Orbit Model works by extrapolation. It is unable to calculate data for any orbit that precedes the earliest entry in the database table. Data for the Orbit Model is specified in the Orbit Model ODL file, which is read only if the PGE requires orbital information.

Orbital path is the path of the satellite over the Earth. It is a number from 0-233 that indicates the region of the Earth covered by a particular orbit. Note that because of the implementation of Orbital Path, there needs to be a mapping between the orbital path calculated by PDPS and the orbital path number expected by the PGEs.

Runtime parameters can be set to values associated with Orbital Processing. The following list of orbital parameters can be placed under runtime parameters:

- Orbit Number.
    - The number of the orbit (starting from zero) and continually increasing.
- Orbital Path Number.
    - The number of the path that maps to the orbit number.
    - The orbital path number is the 0-233 orbital path traversed by the satellite.
- Orbit Number within the Day.
    - The number of the orbit within the given day.
    - It includes any orbit that ends within the given day.
- Granule Number within the Orbit.
    - The number of the granule within a given orbit.
    - It includes any granule that starts within the given orbit.

Figure 11.2.9-1 provides an illustration of the Orbital Processing Production Rule. The PGE in the diagram takes a two-hour input, but is scheduled based on the orbit time and period of the satellite. PDPS uses the data collected at SSI&T to predict the time of the orbit and performs the query to the Science Data Server for the input based on that extrapolated or interpolated orbital time. Granules of input data are allocated to DPRs based on their ability to cover the time period relevant to the DPR.

In the example shown in Figure 11.2.9-1 the length of an orbit is less than the period of the two-hour input, so sometimes a single granule may cover the input time range of a PGE execution

and at other times two granules are required. The production rule would work equally well if the data were of a shorter period (e.g., 1/2 hour) than the orbit of a satellite (e.g., 90 minutes). In such a case three granules would be staged for every execution of the PGE.

The Orbital Processing Production Rule is based (at least for the PGE science metadata ODL file) on the same fields used for the Basic Temporal Production Rule. However, the values specified for the parameters provide orbit information rather than time-period information.
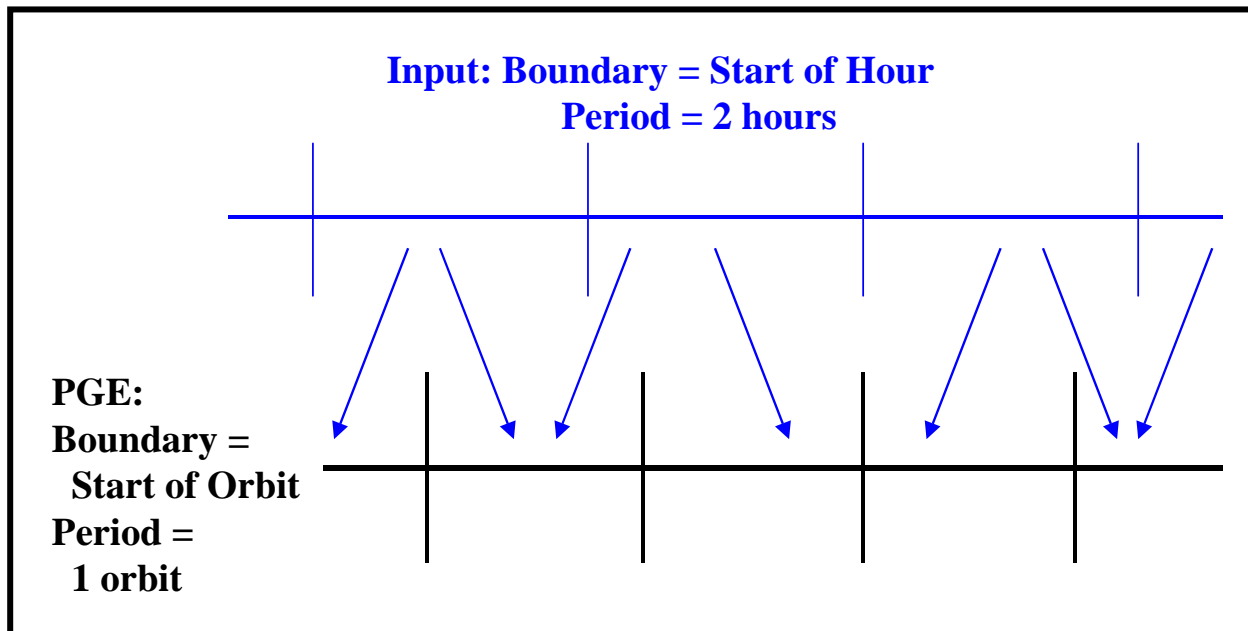
Figure 11.2.9-1. Example of the Orbital Processing Production Rule

11.2.9.1   PGE Science Metadata ODL File Parameters

The following parameters must be set in the PGE science metadata ODL file in order to implement the Orbital Processing Production Rule:

- PLATFORM.
- SCHEDULE_TYPE.
- PROCESSING_PERIOD.
- PROCESSING_BOUNDARY.

The PLATFORM parameter is the name of the platform (satellite) for which the PGE is processing data. Information concerning the orbits of a satellite is stored in the PDPS database. Values that can be assigned to the parameter are subject to the following constraints:

- The string specified can have no more than 25 characters.
- The string specified should match the string specified in the orbit science metadata ODL file. If no matching file is found, an error is reported during SSI&T.

611-EMD-001, Rev. 02

The SCHEDULE_TYPE parameter describes the type of scheduling that is required for the PGE. "Orbit" is the value used for Orbital Processing. As a result, the PGE is scheduled based on the start time and period of the satellite's orbit. Note that PROCESSING_PERIOD and PROCESSING_BOUNDARY must be set correspondingly.

The PROCESSING_PERIOD is the time interval for the data that the PGE processes. Assuming no combination of production rules that would affect the period, data are acquired for the specified PROCESSING_PERIOD and output data are planned for the given period. The value assigned to PROCESSING_PERIOD is of the format "<Period Type>=<Length of Period>". The "Period Type" applicable to the Orbital Processing Production Rule is "ORBITS". For example, "ORBITS=1" would be applied to a PGE that processes data related to one orbit's worth of data.

The PROCESSING_BOUNDARY is the boundary (starting point in time) of the PGE. It specifies when each instance of the PGE should start. Note that the PROCESSING_BOUNDARY and PROCESSING_PERIOD are used in conjunction when scheduling the PGE. Consequently, "START_OF_ORBIT" is the acceptable PROCESSING_BOUNDARY for the Orbital Processing Production Rule. It indicates that the PGE processes data related to each satellite orbit. It must be used in conjunction with a PROCESSING_PERIOD that specifies a "Period Type" of "ORBITS".

11.2.9.2 Orbit Science Metadata ODL File Parameters

The following parameter must be set in the orbit science metadata ODL file in order to implement the Orbital Processing Production Rule:

- PLATFORM.

In addition, the following ODL object is used in defining orbits for the Orbital Processing Production Rule:

- ORBIT_MODEL object.

The value assigned to the PLATFORM parameter in the orbit science metadata ODL file must be exactly the same as that specified for the same parameter in the PGE science metadata ODL file.

The ORBIT_MODEL object is an ODL object that surrounds each orbit definition. An OBJECT/END_OBJECT pair (as shown in the example that follows) is needed for each orbit that is to be expressly defined. PDPS extrapolates or interpolates orbits that are not specifically defined within the file.

```
OBJECT = ORBIT_MODEL
  CLASS = 1
  ORBIT_NUMBER = 1000
  ORBIT_PATH_NUMBER = 68
  ORBIT_PERIOD = "MINS=98"
  ORBIT_ START = "09/21/1999 14:50:00"
END_OBJECT = ORBIT_MODEL
```

The following parameters are set in the ORBIT_MODEL object in order to implement the Orbital Processing Production Rule:

- CLASS.
- ORBIT_NUMBER.
- ORBIT_PATH_NUMBER.
- ORBIT_PERIOD.
- ORBIT_START.

CLASS is a simple counter used to differentiate the different ORBIT_MODEL objects within the file. Each ORBIT_MODEL object needs to have a different CLASS value.

ORBIT_NUMBER is simply the number of the orbit being specified. Each orbit of the satellite has a sequential number associated with it. This is the integer value of the orbit number for the orbit being defined in the ORBIT_MODEL object.

ORBIT_PATH_NUMBER is value of the path for the specified orbit. The orbital path is a number from 0-233 that repeats every 16 days. This is the integer value of the orbital path number for the orbit being defined in the ORBIT_MODEL object.

ORBIT_PERIOD is the length of time it takes for the satellite to complete one orbit. The value assigned to ORBIT_PERIOD has the format "<Period Type>=<Length of Period>" (e.g., "MINS=98"). Note that the "Length of Period" is specified as a positive integer only.

Period Type values for the orbit model science metadata ODL file are:

- "WEEKS"
  - Orbit spans some number of weeks.
  - For example, "WEEKS=2" would be an orbit that takes two weeks to complete.
- "DAYS"
  - Orbit spans some number of days.
  - For example, "DAYS=5" would be an orbit that takes five days to complete.
- "HOURS"
  - Orbit spans some number of hours.
  - For example, "HOURS=4" would be an orbit that takes four hours to complete.
- "MINS"
  - Orbit spans some number of minutes.
  - For example, "MINS=85" would be an orbit that takes eighty-five minutes to complete.
- "SECS"
  - Orbit spans some number of seconds.
  - For example, "SECS=7200" would be an orbit that takes 7200 seconds (two hours) to complete.

ORBIT_START is the start date and time for the orbit defined by the particular ORBIT_MODEL object. Its format is either "MMM DD YYYY HH:MM:SS" or "MM/DD/YYYY HH:MM:SS".

11.2.10 Multiple DPRs for Insertion Time Production Rule

The Multiple DPRs for Insertion Time production rule allows the creation of DPRs for multiple granules with the same insertion time. It affects ASTER L1B routine processing only.

The Multiple DPRs for Insertion Time production rule is implemented when the Production Planner creates a production request using the Production Request Editor. The Production Planner enters the Duration information (Begin and End date and time) for the Insertion Time (versus Collection Time) and then ensures that the **Multiple DPRs** toggle button is depressed. If the **Multiple DPRs** toggle button is not depressed, the default rule is applied and when the production request is saved a DPR is created for the most recent granule with the specified insertion time.

The Multiple DPRs for Insertion Time production rule applies to the key input (e.g., ASTER L1B) only. It does not affect any ancillary data types (for which the most recent granule is selected).

When the Multiple DPRs for Insertion Time production rule is invoked, a DPR is generated for each key-data-type granule returned from the insertion-time query. Therefore, if multiple key-data-type granules are returned for the same collection period, multiple DPRs are generated. The DPR IDs are different in the last three characters, which are generated randomly.

11.2.11 Tiling Production Rule

The Tiling Production Rule allows a PGE to run over a series of specific geographic locations called "tiles". The tiles are defined before the PGE is scheduled, specifying the longitude and latitude of four points that outline each tile. When the PGE is scheduled, it is scheduled for an entire day, and data is queried based on both a timeframe and the geographic location specified. Each run of the PGE for that day is for a specific tile, and only data that overlap or fit within the geographical coordinates of the tile are staged for the PGE.

- Example:
  - A MODIS PGE is designed to run on data for a specific geographic location every day.
  - The location is expressed as a polygon defined by latitude and longitude coordinates.
  - The MODIS PGE is scheduled every day, and data are retrieved that match the time period (the day for which the PGE is being executed) and some part of it falls within the geographic constraints of the tile.
  - The PGE runs and produces data that define information about the particular tile.

Period and **boundary** are used to specify the timing of input data and provide indications of how often the PGE should be executed. But at least some of the input data are retrieved on the basis of the coordinates defined for the tile on which the PGE is executing. In fact there are really two kinds of tiling:

- The PGE takes in data based on geographic shapes (tiles) and produces an output or outputs for the specified geographical coverage.

- The PGE takes in an already tiled product as input.
    − This form of tiling is more like a Metadata Query using a runtime parameter value to acquire the correct tiled data.

There are some possible future enhancements to the Tiling Production Rule but they have not been scheduled yet.

- **Zonal Tiling** supports tiles that cover a band around the Earth between two given latitudes.
- **Tile Clustering** involves grouping tiles that cover nearby geographic locations together so that data that span the tiles may be staged only once.
    − Intended to improve the performance of Tiling.
    − Also provides for the ability to prioritize one group of tiles over others (so specific geographic outputs are produced before other geographic outputs).

Runtime parameters can be set to the ID of the tile being processed. Since PDPS schedules a Tiling PGE to run once per tile, it can pass the identifier of the tile to the PGE. The identifier can be placed under a specified runtime parameter in the PCF, or it can be used in a Metadata Query for a PGE that would use already tiled data as input.

Figure 11.2.11-1 provides an example of the Tiling Production Rule. The PGE runs once per defined tile. So for every tile in the Tile Scheme a Data Processing Request is created to run using data that match the geographic extent of the tile. The PDPS sends the coordinates of the tiles (e.g., Tiles 1 through 3 in Figure 11.2.11-1) to the Science Data Server when requesting data and acquires only the granules that fall fully or partially within the defined tile.

The PGE itself must be set up to handle the fact that the entire area of the tile may not be covered by available data. In addition, because PDPS does not keep track of tiles once they have been produced, the PGE must set the metadata of the output products so a downstream Tiling PGE can acquire the correct granules for a given tile. The PDPS matches up the granules needed for a downstream PGE via a query to the Data Server Subsystem.

11.2.11.1 Tiling Based on Already Tiled Data

As previously stated the second form of Tiling concerns PGEs based on tiles that have already been created by other PGEs. Tiling based on already tiled data is really a combination of the Metadata Query Production Rule and the Tiling Production Rule. The latter is used in running the PGE(s) once per tile, just like any other Tiling PGE. The Metadata Query Production Rule is used in acquiring the previously tiled data by querying the Science Data Server for metadata that match the tile ID that is currently being executed. The query depends on the **runtime parameters** function of Tiling to provide the tile ID relevant to the PGE that is currently being executed.
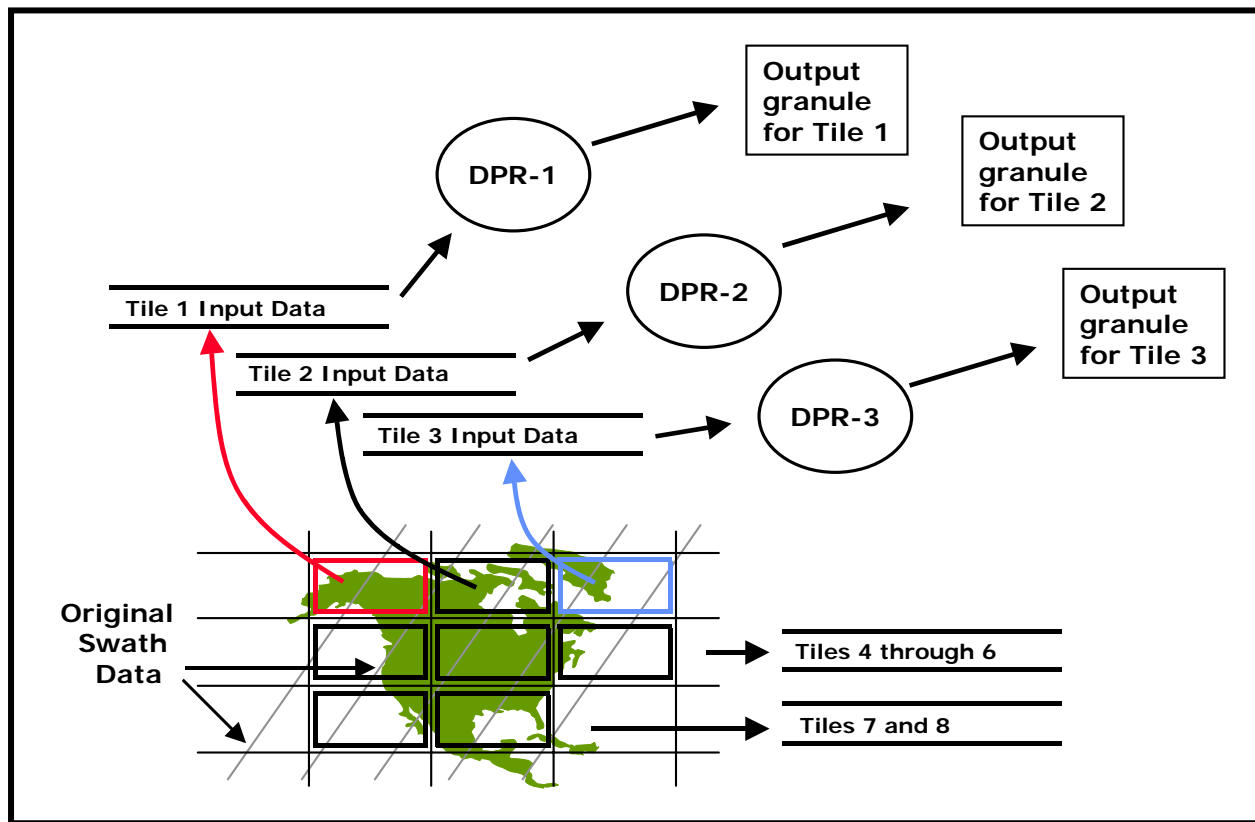
Figure 11.2.11-1. Example of the Tiling Production Rule

The Tiling Production Rule is based (at least for the PGE science metadata ODL file) on the same fields used for the Basic Temporal Production Rule. A PGE that performs Tiling still needs a **boundary** and **period** and other such parameters. The difference is that values specified for some of the fields provide Tiling information. Furthermore, Tiling requires that a tile scheme be identified in the PGE science metadata ODL file. The tile scheme is defined in a tile science metadata ODL file.

11.2.11.2 PGE Science Metadata ODL File Parameters

The following parameters must be set in the PGE science metadata ODL file in order to implement the Tiling Production Rule:

- SCHEDULE_TYPE.
- TILE_SCHEME_NAME.

In addition, the following parameter is used within a PCF_ENTRY when defining the Tiling Production Rule:

- QUERY_TYPE.

The SCHEDULE_TYPE parameter defines the type of scheduling that will be done for the PGE. Values for the Tiling Production Rule are:

- "Tiling"
  - Tile-Scheduled.
  - The PGE is scheduled based on the specified PROCESSING_PERIOD and PROCESSING_BOUNDARY, but a DPR is created for each defined tile.

The TILE_SCHEME_NAME parameter is the name of the Tile Scheme to be used by PDPS when scheduling and executing PGEs for each defined tile. There must be a tile ODL file that matches the specified scheme name.

The QUERY_TYPE parameter specifies the type of query to be performed on the input defined by the PCF_ENTRY Object. It uses the following syntax:

> OBJECT = PCF_ENTRY
> .
> .
> .
> QUERY_TYPE =
> .
> END_OBJECT = PCF_ENTRY

For Tiling PGEs there are two possible values for QUERY_TYPE:

- "Tile"
  - The data for the input are acquired on the basis of the spatial constraints of the current tile.
  - Used for a PGE that takes in raw data and produces one or more tiles of data.
- "Already Created Tile"
  - The input is a tiled output of another Tiling PGE.
  - Used for a PGE that takes input from one or more other Tiling PGEs.
  - A Metadata Query must be added to this PCF_ENTRY in order for the correct tiled input to be acquired.

11.2.11.3 Tile Science Metadata ODL File Parameters

The following parameter must be set in the Tile science metadata ODL file in order to implement the Tiling Production Rule:

- TILE_SCHEME_NAME.

In addition, the following ODL objects are used within a PCF_ENTRY to define the Tiling Production Rule:

- TILE object.
- TILE_COORDINATE object.

The TILE_SCHEME_NAME parameter identifies the tile scheme for which the tile information is being specified. Values are limited by the following constraints:

- The string specified can be no more than 20 characters.

- The string specified should match the string specified for TILE_SCHEME in the PGE science metadata ODL file.

The TILE object is an ODL object that surrounds each tile definition. An OBJECT/END_OBJECT pair (as shown in the example that follows) is needed for each tile that is going to be expressly defined:

```
OBJECT = TILE
 .
 .
 .
END_OBJECT = TILE
```

The following parameters are set in the TILE object in order to implement the Tiling Production Rule:

- CLASS.
- TILE_ID.
- TILE_DESCRIPTION.

CLASS is a simple counter used to differentiate the different TILE objects within the file. Each TILE object needs to have a different CLASS value.

TILE_ID is the tile identifier for the tile being defined. The TILE_ID must be an integer (e.g., TILE_ID = 12) and must be greater than zero but less than the maximum integer. If a Tile ID is defined in other tile schemes, it must have the same coordinates and description.

TILE_DESCRIPTION is a string of characters (255 characters maximum) that describes what the tile is for, such as its geographic location or area that it covers (e.g., TILE_DESCRIPTION = "Upper North America").

The TILE_COORDINATE object is an ODL object that defines a coordinate (latitude and longitude) for a tile. An OBJECT/END_OBJECT pair is needed for each coordinate that is defined. Each tile must have four TILE_COORDINATE objects defined. (Currently only four-sided polygons are allowed; however, a possible future enhancement would provide for polygons with more than four points.) Coordinate objects must follow a clockwise sequence so that if lines were drawn between the points in the order they are given the desired shape would be drawn.

Coordinate objects conform to the following format:

```
OBJECT = TILE
 .
 .
 .
 OBJECT = TILE_COORDINATE
   CLASS =
   LATITUDE =
   LONGITUDE =
 END_OBJECT = TILE_COORDINATE
```

.
.
.
END_OBJECT = TILE

The following parameters are set in the TILE_COORDINATE object in order to implement the Tiling Production Rule:

- CLASS.
- LATITUDE.
- LONGITUDE.

The CLASS parameter (e.g., CLASS = 1) is an object counter that is used only to distinguish objects. The value assigned to CLASS must be an integer greater than zero and must be unique in the file for the particular type of object.

The LATITUDE parameter (e.g., LATITUDE = 12.15) describes the latitude component of the tile coordinate. There is one LATITUDE entry per TILE_COORDINATE object.

The LONGITUDE parameter (e.g., LONGITUDE = -43.22) describes the longitude component of the tile coordinate. There is one LONGITUDE entry per TILE_COORDINATE object.

11.3 Intermittent Activation

The conditions for executing most PGEs are well defined. The most common activation condition is the availability of all input data sets. Similarly, the frequency of execution is usually well defined (e.g., run once for every granule or run monthly averages once a month). However, some PGEs have additional or different constraints on when they are run.

A PGE can be set up to run on every n^{th} instance of input data. For example, a QA PGE that is run on a daily product may need to be run only every fifth day to provide a spot check. Note that this does **not** refer to the common case of running a weekly averaging PGE only once each week, which would be handled by the Basic Temporal Production Rule and the time ranges specified for the input and output ESDTs. Rather, this is a special case where a PGE **can** be run every day (or hour, week, etc.), but for some reason (such as a QA check) it is desired to run the PGE only every n^{th} day.

To implement Intermittent Activation the Production Planner supplies the following information (via the Production Request Editor) when creating a production request:

- **Number to Skip**
  - Number of DPRs to be skipped (not executed).
  - Entered in the **Skip** field on the Production Request Editor.
- **Number to Keep**
  - After skipping the specified number of DPRs, how many are to be kept?
  - Entered in the **Keep** field on the Production Request Editor.
  - The number to keep is usually one but could be any number.
- **Skip First**
  - Button on the Production Request Editor.

- Selected to skip the first DPR.
- Not selected if the first DPR is to be run.

The Planning Subsystem uses the preceding information to establish a pattern of execution. The pattern is effective for the single PR in which the "number to skip" and the "number to keep" are specified; it is not maintained between PRs.

The following example of Intermittent Activation is illustrated in Figure 11.3-1:

- The Production Planner prepares a production request for a 14-day period, generating 14 DPRs.
- The Production Planner made the following selections on the Production Request Editor:
  - Entered "4" in the **Number to Skip** field.
  - Entered "1" in the **Number to Keep** field.
  - Did **not** select the **Skip First** button.
- Consequently, the following results are obtained:
  - First DPR runs.
  - Four DPRs (second through fifth) are skipped.
  - Sixth DPR runs.
  - Four DPRs (seventh through tenth) are skipped.
  - Eleventh DPR runs.
  - Remaining three DPRs (twelfth through fourteenth) are skipped.
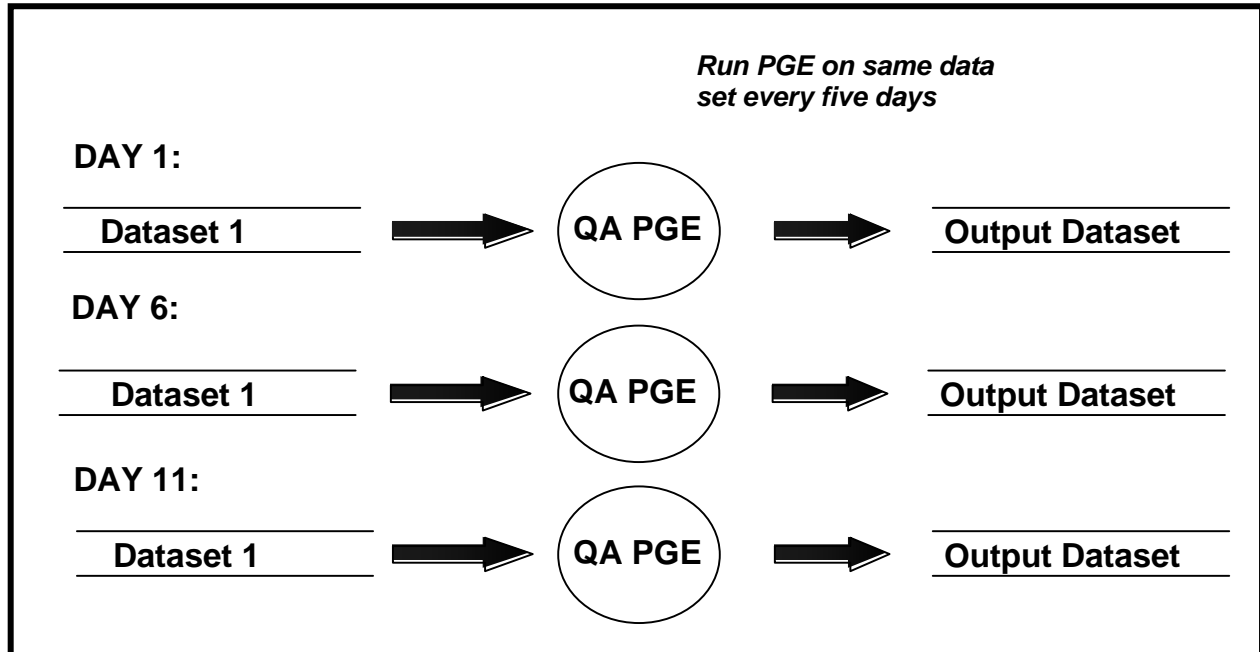
Figure 11.3-1. Example of Intermittent Activation

12. Resource Planning

12.1 Resource Planning Process

The Resource Planning process is the mechanism by which reservations for non-routine ground events are defined and controlled. Such events may include testing, corrective maintenance, preventive maintenance or system upgrades, or any other event that requires DAAC production processing resources. Resource planning defines ground events, which are also used in production planning; thus, resource planning can take place whenever a production plan needs to be created. In general, this will occur on a biweekly basis for 30-day plans, on a weekly basis for ten-day plans, and on a daily basis. However, ground events can be entered at any time. The important point is that it is necessary to be aware of the anticipated processing load and upcoming maintenance events for about the next month.

Resource Planning includes two general types of activities; i.e., Resource Definition and Resource Scheduling. The site Resource Planner uses the Resource Editor GUI within the Planning Subsystem to define system resources used in production processing. The Resource Planner and Resource Manager use the Resource Scheduler GUI to schedule non-routine events against system resources.

Subsequent sections related to Resource Planning address the following topics:

- Section 12.2   An overview of the process for defining resources and step-by-step procedures for using the Resource Editor.
- Section 12.3   An overview of the process for scheduling resources and step-by-step procedures for using the Resource Scheduler.
- Section 12.4   An overview of the process for tuning system parameters related to Resource Planning and a description of the process for changing configuration parameters.
- Section 12.5   An overview of the process and step-by-step procedures for troubleshooting Resource Planning problems.

12.2 Defining Resources

The Resource Planner uses the Resource Editor GUI to define system resources used in production processing in the following terms:

- "Disks."
- "Virtual computers" (sets of CPUs and associated memory and disks).
- "Strings" (sets of virtual computers).
- "Real computers" (hosts that are composed of one or more virtual computers).
- "AutoSys" ("strings" associated with the production processing software).
- Generic "hardware."

The following general process is used for defining production resources:

- Determine what production resources are available.
- Determine the distribution of resources among operating modes.
- Define resources for each mode using the Resource Editor GUI.

Each procedure outlined has an **Activity Checklist** table that provides an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one - *Order* shows the order in which tasks could be accomplished.

Column two - *Role* lists the Role/Manager/Operator responsible for performing the task.

Column three -*Task* provides a brief explanation of the task.

Column four - *Section* provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - **Complete?** is used as a checklist to keep track of which task steps have been completed.

Table 12.2-1, below, provides an Activity Checklist for Defining Resources.

Table 12.2-1. Defining Resources - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner	Log in to System Hosts	(P) 12.2.1	
2	Resource Planner	Launch the Resource Editor	(P) 12.2.2	
3	Resource Planner	Determine Actual Processing Resources	(P) 12.2.3	
4	Resource Planner	Add a Resource	(P) 12.2.4	
5	Resource Planner	Add/Modify a Disk	(P) 12.2.4.1	
6	Resource Planner	Add/Modify a Virtual Computer	(P) 12.2.4.2	
7	Resource Planner	Add/Modify a Real Computer	(P) 12.2.4.3	
8	Resource Planner	Add/Modify a String	(P) 12.2.4.4	
9	Resource Planner	Add/Modify an Autosys Resource	(P) 12.2.4.5	
10	Resource Planner	Add/Modify a Hardware Resource	(P) 12.2.4.6	
11	Resource Planner	Modify a Resource	(P) 12.2.5	
12	Resource Planner	Delete a Resource	(P) 12.2.6	
13	Resource Planner or DAAC Staff	Shut Down Resource Definition Applications	(P) 12.2.7	

12.2.1 Log in to System Hosts

Logging in to system hosts is accomplished from a UNIX command line prompt. Table 12.2-2 presents (in a condensed format) the steps required to log in to system hosts. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   At the UNIX command line prompt enter:

setenv DISPLAY <client name>:0.0

- Use either the X terminal/workstation IP address or the machine-name for the client name.
- When using secure shell, the DISPLAY variable is set just once, before logging in to remote hosts. If it were to be reset after logging in to a remote host, the security features would be compromised.

2   In the terminal window (at the command line prompt) start the log-in to the appropriate host by entering:

/tools/bin/ssh <host name>

- The **-l** option can be used with the ssh command to allow logging in to the remote host (or the local host for that matter) with a different user ID. For example, to log in to x0pls02 as user cmops enter:

  /tools/bin/ssh -l cmops x0pls02

- Depending on the set-up it may or may not be necessary to include the path (i.e., /tools/bin/) with the ssh command. Using ssh alone is often adequate. For example:

  ssh x0pls02

  - or -

  ssh -l cmops x0pls02

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- Examples of Science Processor host names include **e0spg11**, **g0spg11,** and **l0spg11**.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- Examples of Sun internal server host names include **e0acs06**, **g0acs06**, and **l0acs06**.
- Examples of Access/Process Coordinators (APC) Server host names include **e0acg11**, **g0acg01**, and **l0acg02**.
- Examples of Ingest Server host names include **e0icg11, g0icg01**, and **l0acg02**.
- Examples of Sun external server host names include **e0ins01, g0ins01**, and **l0ins01**.
- If you receive the message, "Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?" enter **yes** ("y" alone will not work).

- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter** *passphrase* **for RSA key '**<*user@localhost*>**'** appears; continue with Step 3.
- If you have not previously set up a secure shell passphrase, go to Step 4.

3   If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, enter:

<passphrase>

- If a command line prompt is displayed, log-in is complete.
- If the passphrase is unknown, press **Return/Enter**, which should cause a **<user@remotehost>'s password:** prompt to appear (after the second or third try if not after the first one), then go to Step 4.
- If the passphrase is entered improperly, a **<user@remotehost>'s password:** prompt should appear (after the second or third try if not after the first one); go to Step 4.

4   If a prompt for **<***user@remotehost***>'s password:** appears, enter:

<password>

- A command line prompt is displayed.
- Log-in is complete.

Table 12.2-2. Log in to System Hosts - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**setenv DISPLAY <client name>:0.0**	**enter text, press Enter**
2	**/tools/bin/ssh <host name>** (as applicable)	**enter text, press Enter**
3	**<passphrase>** (if applicable)	**enter text, press Enter**
4	**<password>** (if applicable)	**enter text, press Enter**

12.2.2   Launch the Resource Editor

The Resource Editor is invoked from a UNIX command line prompt. Table 12.2-3 presents (in a condensed format) the steps required to launch the **Resource Editor** GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 12.2.1).

2   In the terminal window, at the command line, enter:

    cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
    - TS1   - Science Software Integration and Test (SSI&T)
    - TS2   - New Version Checkout
    - OPS   - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3   Set the application environment variables by entering:

    setenv ECS_HOME /usr/ecs/

- Application home environment is entered.
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4   Start the Resource Planning background processes by entering:

    EcPlRpAllStart <MODE> <application_id>

- The Resource Planning background processes are launched.
- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a resource planning session.

5   Start the **Resource Editor** GUI by entering:

    EcPlRpReStart <MODE> <application_id>

- The **Resource Editor** GUI is launched.
- The **Resource Editor** GUI displays a list of defined resources and a series of buttons that enable the following operations:
    - **New...**   Add a resource definition. (Section 12.2.4)
    - **Modify...**   Edit or review the details of an existing resource definition. (Section 12.2.5)
    - **Delete**   Delete a resource definition. (Section 12.2.6)
    - **Fetch Baseline**   [not used]
    - **Load Baseline**   [not used]

Table 12.2-3. Launch the Resource Editor - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables	**enter text, press Enter**
4	**EcPlRpAllStart <MODE> <application_id>**	**enter text, press Enter**
5	**EcPlRpReStart <MODE> <application_id>**	**enter text, press Enter**

12.2.3 Determine Actual Processing Resources

The Resource Editor allows the authorized operator to define resources in the following categories:

- **Disks:** Disk partitions that are associated with and provide temporary data storage for the input and output files used in processing.
- **Virtual Computers:** Virtual computers composed of CPUs, random-access memory (RAM), and associated-disk(s). The CPUs and RAM specified for a virtual computer are components of the real computer from which the virtual computer is derived.
- **Strings:** Sets of one or more virtual computers. Strings are associated with the processing software (AutoSys). A dual science processor configuration can be defined by specifying strings containing virtual computers derived from different real computers.
- **Real Computers:** Physical computing devices (hosts), each of which contains one or more CPUs. Each science processor host ("real" computer) is divided into one or more virtual computers by allocating CPUs and RAM from the real computer to the virtual computer(s).
- **AutoSys:** Identifies the string(s) of virtual computers used by the production processing software.
- **Hardware:** Any type of equipment that is not defined as a computer or disk may be defined as "hardware."

Resources at the DAACs need to be allocated among the operating modes. For example, disk partitions at a DAAC could be split among the operating modes as follows:

- OPS – 60%.
- TS1 - 20%.
- TS2 - 20%.

Actual percentages would be determined at each DAAC and it would be advantageous to reserve some nominal percentage of the disk space (e.g., two to five percent) as a safety buffer. In any case, it is critical to ensure that the sum of the disk space assigned to the various modes is no more than the total disk space available.

CPUs and RAM need to be allocated among modes in the same manner. However, it is not necessary to be exact with the CPU count or RAM amount.

- There is no one-to-one mapping of CPU allocation with actual CPUs on the science processor.
- Actual CPU usage during processing is limited by the operating system (OS).
    - If ten CPUs have been specified for a particular mode, only ten Data Processing Requests (DPRs) can be running the Execute job at a given time.
    - What is really being defined is the maximum number of DPRs that will execute at a given time.
- It is important to monitor the load on each science processor.
    - CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs on each science processor.
    - If monitoring indicates that the processor is underused when OPS mode is at full processing capacity, the number of CPUs allocated to OPS mode could probably be increased.
    - If the science processor is at full capacity when OPS mode is at full processing capacity and it is suspected that the processor may be overworked, the number of CPUs allocated to OPS mode should be reduced.
- Random-access memory (RAM) is subject to the same considerations as CPUs.
    - RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor.
- The OS takes care of true CPU and RAM allocation.

Before data processing resources can be defined, it is necessary to know what resources are actually available at the DAAC. Some resources are defined in terms of other resources; for example, a string is defined as one or more virtual computers. However, it is generally necessary to have the following types of information available in order to define processing resources:

- Host names ["real computers"].
- Number of processors [CPUs] available on each host.
- Operating System (OS) for each host.
- Memory [RAM] on each host.
- Total disk space.
- AutoSys instance(s) at the DAAC.

Table 12.2-4 presents (in a condensed format) the steps required to determine the actual processing resources to be defined. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

NOTE: The procedure to determine the actual processing resources to be defined starts with the assumption that the DISPLAY environment variable has been set (Refer to Section 12.2.2).

1   Access a terminal window logged in to the applicable Science Processor host.
        • Examples of Science Processor host names include **e0spg11**, **g0spg11,** and **l0spg11**.
        • For detailed instructions refer to the **Log in to System Hosts** procedure
          (Section 12.2.1).

2   To access the mount point enter:

        cd /usr/ecs/<MODE>/CUSTOM/pdps/<processor>/data/DpPrRm/<processor>_disk

        • Change directory to the disk mount point (e.g.,
          /usr/ecs/OPS/CUSTOM/pdps/e0spg11/data/DpPrRm/e0spg11_disk).

3   To determine disk size and usage enter:

        df -k . (being sure to include the dot)

        • Information concerning disk size and use is displayed; for example:

          Filesystem   Type kbytes   use   avail %use Mounted on
          /dev/dsk/xlv/vgo1   xfs 413394688 164646048 248748640 40 /vol1

        • In the preceding example the total disk space is 413,394,688 kilobytes or
          413,394.69 megabytes (413 gigabytes).

4   To obtain Information concerning the number of CPUs and amount of RAM (memory)
        enter:

        hinv

        • The hinv command is available on Silicon Graphics, Inc. (SGI) hosts only.
        • Information concerning CPUs and RAM (memory) is displayed; for example (not all
          rows are shown):

          Processor 0: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.6
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0
          Processor 1: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.6
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0
          Processor 2: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.6
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0
          Processor 3: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.6
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0
          Processor 4: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.5
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0
          Processor 5: 194 MHZ IP25
          CPU: MIPS R10000 Processor Chip Revision: 2.5
          FPU: MIPS R10010 Floating Point Chip Revision: 0.0

Secondary unified instruction/data cache size: 1 Mbyte
Data cache size: 32 Kbytes
Instruction cache size: 32 Kbytes
Main memory size: 2048 Mbytes, 8-way interleaved
[…]

- .

5   Repeat Steps 1 through 4 for all other science processors (if any).

NOTE: Steps 6 through 12 describe the use of the Netscape browser to determine certain types of information concerning computer resources (including the number of CPUs and amount of RAM), which can be determined using the **hinv** command as described in Step 4. However, the "as-built" file accessed using the Netscape browser lists the necessary operating system information in addition to CPU and RAM data. The advantage of the **hinv** command is that it provides real-time data and is reliably up to date. The advantage of the "as-built" file accessed using the Netscape browser is that it provides operating system data that is not available using the **hinv** command.

6   To launch the Netscape web browser enter:

netscape &

- It may be necessary to change directories before launching the Netscape web browser (e.g., cd /tools/bin/netscape3.01).
- The Netscape web browser is displayed.

7   In the browser's **Location** (**Go To**) field enter the following address:

http://cmdm-ldo.raytheon.com/baseline/

- The EMD Baseline Information System web page is displayed.

8   **Single-click** on the **EMD Configuration** link.
- A table of files is displayed.

9   **Single-click** on the **Asbuilts** link for the relevant DAAC.
- A list of files is displayed.

10   **Single-click** on the file name corresponding to the desired host.
- For example, the file name for host x0spg01 would be x0spg01.asbuilt.html.
- A report containing the following types of information (among other items) is displayed:
  – **Host Name** ["real computer"].
  – **Processors** [CPUs].
  – **Operating System.**

- **Memory** [RAM].
- **Interrogation Date** (useful in determining how up-to-date the information is).

11   **Single-click** on the browser **Back** button.
- The list of "as-built" files is displayed.

12   Repeat Steps 10 and 11 for all other science processors (if any).

13   Exit from the Netscape browser when the necessary information has been acquired by executing the following menu path:

File → Exit

- The Netscape browser disappears.

14   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 12.2.1).

15   To gain access to the directory containing the AutoSys configuration files enter:

cd /usr/ecs/<MODE>/COTS/<autotree>/autouser

- Change directory to the directory (e.g.,
  /usr/ecs/<MODE>/COTS/autotreeb/autouser,
  /usr/ecs/<MODE>/COTS/autotree/autouser,
  /data1/SHARED/COTS/autotree/autouser) containing the set-up files (e.g.,
  FMR.autosys.csh.g0sps06) and the AutoSys configuration files (e.g., **config.FMR**).
- The particular path to be typed may vary from site to site.
- The AutoSys instance at the DAAC is identified by three capital letters appended to the beginning of the set-up files and the end of the configuration file.
  - Typically, AutoSys instances at the DAACs are identified as **FMR**.
- It is possible to have multiple AutoSys instances installed at a DAAC.

Table 12.2-4. Determine Actual Processing Resources - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	UNIX window (Science Processor)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/pdps/<processor> /data/DpPrRm/<processor>_disk**	**enter text, press Enter**
3	**df -k .** (being sure to include the dot)	**enter text, press Enter**
4	Observe the disk capacity	**read text**
5	**hinv**	**enter text, press Enter**

Step	What to Enter or Select	Action to Take
6	Observe the number of CPUs and total memory (RAM)	**read text**
7	Repeat Steps 1 through 6 for all other science processors (if any)	
8	Launch Netscape	**enter text, press Enter**
9	**http://cmdm-ldo.raytheon.com/baseline/**	**enter text, press Enter**
10	**EMD Configuration** link	**single-click**
11	**Asbuilts** link (for the relevant DAAC)	**single-click**
12	**<file name>** (corresponding to the desired host)	**single-click**
13	Observe the number of CPUs, total memory (RAM), and Operating System identification	**read text**
14	**Back** button	**single-click**
15	Repeat Steps 12 through 14 for all other science processors (if any)	
16	**File → Exit** (to exit from Netscape)	**single-click**
17	UNIX window (Queuing Server)	**single-click** or use procedure in Section 12.2.1
18	**cd /usr/ecs/<MODE>/COTS/<autotree>/autouser**	**enter text, press Enter**
19	Observe the identification of the AutoSys instance	**read text**

12.2.4 Add a Resource

These procedures address adding such resources as computers, disk partitions, strings, and generic hardware resources to the resource planning list. Resources at the DAACs need to be allocated among the operating modes. For example, disk partitions at a DAAC could be split among the operating modes as follows:

- OPS – 60%.
- TS1 - 20%.
- TS-2 - 20%.

Actual percentages would be determined at each DAAC and it would be advantageous to reserve some nominal percentage of the disk space (e.g., two to five percent) as a safety buffer. In any case, it is critical to ensure that the sum of the disk space assigned to the various modes is no more than the total disk space available.

Table 12.2-5 presents (in a condensed format) the steps required to add a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).
- The **Resource Editor** GUI is displayed.

2   From the **Resource Editor** GUI, select the type of resource to be added from the list on the **Resource Type** button.

3   Either **single-click** on the **New…** button or execute the following menu path:

File → New

4   Define the resource as specified in the corresponding procedure section.
- Refer to the specified section for defining the desired type(s) of resources:
  - **Disk** – Section 12.2.4.1.
  - **Virtual Computer** – Section 12.2.4.2.
  - **Real Computer** – Section 12.2.4.3.
  - **String** – Section 12.2.4.4.
  - **Autosys** – Section 12.2.4.5.
  - **Hardware** – Section 12.2.4.6.
- Resources should generally be added in the preceding order (due to dependencies among resources).

5   After the data have been entered, **single-click** on one of the following buttons:
- **Save** to save **and** exit.
- **Cancel** to exit without saving.

Table 12.2-5. Add a Resource - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	**Resource Type** option button	**single-click**
3	**New…** button	**single-click**
4	Make entries in the necessary fields	Use procedures in Sections 12.2.4.1 through 12.2.4.6
5	**Save** button	**single-click**

12.2.4.1 Add/Modify a Disk

1   Enter the relevant information in the following fields on the **Disk Details** GUI:
- **Resource Name**   Operator-defined name for the resource. (Required)
  - Example: e0spg11_disk_OPS.
- **Activity**   System-generated default activity; can be changed by clicking on the bar in the **Activity** field and then clicking on one of the available options.

- **Partition Size** The size of the disk partition, in **megabytes**. (Required)
  - Although the label on the GUI implies that partition size should be entered in "blocks," the label is erroneous. Enter the partition size in megabytes.
- **Block Size** Block size in bytes (always 1024) used for the disk. (Required)
- **Comments** Operator comments on the resource.

2   After the data have been entered, **single-click** on one of the following buttons:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.2 Add/Modify a Virtual Computer

1   Enter the relevant information in the following fields on the **Virtual Computer Details** GUI:
- **Resource Name** Operator-defined name for the virtual computer. (Required)
  - Example: e0spg11_vc_OPS.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Number of CPUs** Number of CPUs within the virtual computer. (Required)
- **Total RAM** The total memory for the virtual computer in megabytes. (Required)
- **Operating System** The operating system name/version for the computer. (Required)
- **Disks** A list of the disks previously defined for that site. This list of disks from which to select is used when a disk is associated (or disassociated) with the computer. After items are highlighted, arrow buttons will move items from this list to **Associated Disks** or from the list of **Associated Disks** to the **Disk** list.
- **Associated Disks** Disks in this list are associated with the computer.
- **Comments** Operator comments on the resource.

2   After the data have been entered, **single-click** on one of the following buttons:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.3 Add/Modify a Real Computer

1   Enter the relevant information in the following fields on the **Real Computer Details** GUI:
- **Resource Name** Operator-defined name for the real resource. (Required)
  - Example: e0spg11.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Computers** A list of the virtual computers previously defined for that site. This list of virtual computers from which to select is used when a virtual computer is

associated (or disassociated) with the real computer. After items are highlighted, arrow buttons will move items from this list to **Associated Computers** or from the list of **Associated Computers** to the **Computers** list.

- **Associated Computers**   Virtual computers in this list are associated with the real computer.
- **Comments**   Operator comments on the resource.

2   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.4 Add/Modify a String

1   Enter the relevant information in the following fields on the **String Details** GUI:
- **Resource Name**   Operator-defined name for the resource. (Required)
  - Example: e0spg11_string_OPS.
- **Activity**   System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Computers**   A list of virtual computers previously defined for that site. This list of computers from which to select is used when a computer is associated (or disassociated) with the string. After items are highlighted, arrow buttons will move items from this list to **Associated Computers** or from the list of **Associated Computers** to the **Computer** list.
- **Associated Computers**   Virtual computers in this list are associated with the string.
- **Comments**   Operator comments on the resource.

2   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.5 Add/Modify an AutoSys Resource

1   Enter the relevant information in the following fields on the **Autosys Details** GUI:
- **Resource Name**   Operator-defined name for the AutoSys resource. (Required)
  - Example: FMR.
- **Activity**   System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Strings**   A list of the strings previously defined for that site. This list of strings from which to select is used when a string is associated (or disassociated) with the AutoSys resource. After items are highlighted, arrow buttons will move items from this list to **Associated Strings** or from the list of **Associated Strings** to the **Strings** list.

- **Associated Strings**   Strings in this list are associated with the AutoSys resource.
- **Comments**   Operator comments on the resource.

2   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.6   Add/Modify a Hardware Resource

1   Enter the relevant information in the following fields on the **Hardware Resource Details** GUI:
- **Resource Name**   Operator-defined name for the resource. (Required)
  - – Example: e0spg11_cdrom_OPS.
- **Activity**   System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Comments**   Operator comments on the resource.

2   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.5   Modify a Resource

Table 12.2-6 presents (in a condensed format) the steps required to modify a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).
- The **Resource Editor** GUI is displayed.

2   From the list of resources displayed on the **Resource Editor** GUI, **single-click** on the resource to be modified.

3   **Single-click** on the **Modify…** button to access the appropriate Details GUI.

4   Make the modifications.
- For field descriptions, refer to Sections 12.2.4.1 through 12.2.4.6.

5   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- Cancel to exit without saving.

Table 12.2-6. Modify a Resource - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	Select the resource to be modified	**single-click**
3	**Modify…** button	**single-click**
4	Make the modifications	Use procedures in Sections 12.2.4.1 through 12.2.4.6
5	**Save** button	**single-click**

12.2.6   Delete a Resource

Table 12.2-7 presents (in a condensed format) the steps required to delete a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).
- The **Resource Editor** GUI is displayed.

2   From the list of resources displayed on the **Resource Editor** GUI, **single-click** on the resource to be deleted.

3   **Single-click** on the **Delete** button.
- A dialogue box pops up to verify whether the resource is really to be deleted.

4   **Single-click** on one of the following buttons as appropriate:
- **OK** to remove the resource from the list and from the PDPS database **and** exit**.**
- **Cancel** to exit **without** deleting the resource.

Table 12.2-7. Delete a Resource - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	**\<resource\>** (to be deleted)	**single-click**
3	**Delete** button	**single-click**
4	**OK** button	**single-click**

12.2.7   Shut Down Resource Definition Applications

When resource definition activities have been completed, the Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the resource planning applications. If any of the three processes

remains active, it is likely to interfere with subsequent attempts to launch resource planning applications.

Shutting down resource definition applications starts with the assumption that the **Resource Editor** GUI is currently being displayed.

Table 12.2-8 presents (in a condensed format) the steps required to shut down resource definition applications. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   To exit from the **Resource Editor** GUI when resource planning activities have been completed execute the following menu path:

File → Exit

- The **Resource Editor** GUI disappears.

2   After quitting the **Resource Editor** GUI **single-click** in the UNIX window used to start the resource definition applications.

3   Shut down the Message Handler, System Name Server, and Resource Model by entering:

EcPlRpSlayAll <MODE> <application_id>

- The **Message Handler** GUI disappears.

4   To obtain a list of active processes in the specified mode enter:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

   ps -auxwww | grep <MODE>

5   Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.
- None of the following processes should be active:
   – EcPlRpRe
   – EcPlRpSi
   – EcPlRpTl
   – EcPlMsh
   – EcPlSns
   – EcPlRpRm

6   If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 \<process ID1\> [\<process ID2\>] [\<process ID3\>] […]

7   Repeat Steps 4 through 6 as necessary.

Table 12.2-8. Shut Down Resource Definition Applications -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**File → Exit** (to quit the **Resource Editor** GUI)	**single-click**
2	**OK** button	**single-click**
3	UNIX window (Planning/Management Workstation)	**single-click**
4	**EcPlRpSlayAll \<MODE\> \<application_id\>**	**enter text, press Enter**
5	**ps -ef | grep \<MODE\>**	**enter text, press Enter**
6	Identify resource planning process(es) that has (have) not shut down	**read text**
7	**kill -15 \<process ID1\> [\<process ID2\>] [\<process ID3\>] […]** if necessary	**enter text, press Enter**

12.3 Scheduling Resources

The Resource Planner and Resource Manager are both involved in resource scheduling using the Resource Scheduler. The Production Planner and Production Monitor are involved in the implementation of ground events.

- Resource Planner processes resource reservation requests for ground events.
- Resource Manager commits resource reservations.
- Production Planner sends committed resource reservations (ground events) to Data Processing via the Planning Workbench.
- Production Monitor monitors execution of ground events in processing.

The following process is used for generating and implementing resource reservations (ground events):

- Personnel who have a need for Planning Subsystem or Data Processing Subsystem resources submit requests for time on specified resources to accomplish the non-routine activities that they plan to undertake.
  - Depending on DAAC policy, many personnel may have access to the resource planning applications for creating resource reservation requests.
  - Alternatively, personnel may have to contact the Resource Planner to have resource reservation requests entered for them.

- The Resource Planner reviews requests for resource reservations to determine if the requests are valid.
  - Request information includes a description of the activity, the resources required, the time period(s) for using the requested resource(s), comments explaining the variance from normal use.
  - Resource Planner may decide to forward the request to a "sponsor" for validation.
  - A sponsor is someone who evaluates a resource reservation request based on expertise that is particularly relevant to the resource reservation request.
- If the Resource Planner or sponsor determines that the request to reserve the resource is valid, the Resource Planner "approves" it along with all other requests that have been validated.
  - The set of all validated resource reservation requests is considered a draft Resource Plan.
- The scheduling software identifies conflicts (if any) in the draft Resource Plan and alerts the Resource Planner to the problem(s).
- If possible, the Resource Planner resolves all conflicts before presenting the proposed plan to the Resource Manager to have the resources committed.
  - However, during the conflict-resolution process the Resource Planner may have to consult with resource requesters and the Resource Manager to ensure that the reserved resources will not have adverse effects on the DAAC's high-priority events.
- When the Resource Planner has achieved a conflict-free plan, it is presented to the Resource Manager to be implemented.
- The Resource Manager "commits" the resource plan, which signals the Planning Subsystem that the plan can be implemented.
  - Committing a plan actually involves committing all of the individual approved resource reservation requests that collectively make up the plan.
- All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing.
  - Resource reservations/ground events are not sent to data processing until they have been included in a production plan.
  - Refer to Section 13, Production Planning, to see how production plans are created and activated.
- In Data Processing a ground event job for each resource reservation is sent to the specified resource(s) at the indicated start time.
  - If a data processing job is already using the specified resource(s) at the ground event's scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job.

Table 12.3-1, below, provides an Activity Checklist of Resource Scheduling activities.

Table 12.3-1. Resource Scheduling - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner or DAAC Staff	Launch the Resource Scheduler	(P) 12.3.1	
2	Resource Planner or DAAC Staff	Create a Resource Reservation Request	(P) 12.3.2	
3	Resource Planner or DAAC Staff	Edit a Resource Reservation Request	(P) 12.3.3	
4	Resource Planner or Sponsor	Validate or Reject a Resource Reservation Request	(P) 12.3.4	
5	Resource Planner	Approve a Resource Reservation Request	(P) 12.3.5	
6	Resource Manager/Resource Planner	Commit Resource Reservation Requests	(P) 12.3.6	
7	Resource Planner	Review the Resource Timeline	(P) 12.3.7	
8	Resource Planner	Delete a Resource Reservation Request	(P) 12.3.8	
9	Resource Planner or DAAC Staff	Shut Down the Resource Scheduler	(P) 12.3.9	

12.3.1 Launch the Resource Scheduler

The **Resource Scheduler** is invoked from a UNIX command line prompt. Table 12.3-2 presents (in a condensed format) the steps required to launch the **Resource Scheduler**. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 12.2.1).

2   In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1   - Science Software Integration and Test (SSI&T)
  - TS2   - New Version Checkout
  - OPS   - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3   Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered.
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4   Start the Resource Planning background processes by entering:

EcPlRpAllStart <MODE> <application_id>

- The Resource Planning background processes are launched.
- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a resource planning session.

5   Start the **Resource Scheduler** GUI by entering:

EcPlRpSiStart <MODE> <application_id>

- The Resource Scheduler is launched.
- The **Resource Scheduler** Graphical User Interface (GUI) is accessed. The GUI displays the Resource Reservation List, activity type, and a series of buttons that enable the following operations:
    - **New...** Create a resource reservation request. (Section 12.3.2)
    - **Modify...**   Edit or review the details of an existing resource reservation request. (Section 12.3.3)
    - **Approve**   Used to indicate that the resource reservation request(s) has (have) been validated and a draft resource plan can be created. Clicking on this button causes the Planning Subsystem to determine whether there are conflicts between this resource reservation and other reservations. The Planning Subsystem detects conflicts and reports them to the operator. (Section 12.3.5)
    - **Commit globally**   Commit all "approved" resource reservations; at this point the ground events will be accessible by the production planning software (Section 12.3.6); however, resource reservations/ground events are not sent to data processing until they have been included in an activated production plan.
    - **Time Line**   Display a timeline-oriented view of the resource plan. (Section 12.3.7)
    - **Report**   The **Report** option is disabled. The reports have been deleted from the system requirements.

Table 12.3-2. Launch the Resource Scheduler - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables if necessary	**enter text, press Enter**
4	**EcPlRpAllStart <MODE> <application_id>**	**enter text, press Enter**
5	**EcPlRpSiStart <MODE> <application_id>**	**enter text, press Enter**

12.3.2 Create a Resource Reservation Request

Table 12.3-3 presents (in a condensed format) the steps required to create a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2   From the **Resource Scheduler** GUI, **single-click** on the **New...** button to access the **Resource Reservation Request Edit/Definition** GUI.

3   Enter resource request identification information into the displayed fields. Press **Tab** to move from field to field. **NOTE:** Data that is system-generated is identified.
- **Request Name**   Operator-provided name for the resource request. (Required)
- **Edited Date**   System-generated date of request entry.
- **Originator**   Operator-provided name of the authorized user preparing the resource request.
- **Sponsor**   Operator -provided name of the individual who is to review and validate the Resource Request (the subject-matter-expert). (Required)

4   Enter resource scheduling information into the displayed fields. Press **Tab** to move from field to field.
- **Activity Type**   Operator-provided description of the type of activity; selected by the operator from a selection list of valid options. (Required)
- **Priority**   Operator-provided priority for the activity. Use the slider to select the appropriate priority on a scale from 0 to 100. One (1) denotes the highest priority and 100 designates the lowest.
- **Description**   Operator-provided description of the activity for which the resource is required. (Required)
- **Resource...**   See Section 12.3.2.1, below. (Required)
- **Interval...**   Not applicable to new resource reservation requests; may be applicable when editing a resource reservation request. See Section 12.3.3.1, below.

5   Enter duration information into the displayed fields to define the period over which the resource is required. Press **Tab** to move from field to field.
- **Start Date**   Operator-provided start date of the resource request period. Enter in <**MM/DD/YYYY>** format. (Required)
- **Start Time**   Operator-provided start time of the resource request. Enter in <**hh:mm:ss>** format. (Required)
  - **Start Time** must be later than the time when the resource reservation request will be saved; otherwise, it will not be possible to save the request.
- **Stop Date**   Operator-provided stop date of the resource request period. Enter in <**MM/DD/YYYY>** format. If a reservation is to be repeated over some **frequency** (see below), the stop date specifies the end date in the date range of the reservation request. (Required)
- **Stop Time**   Operator-provided stop time of the resource request. Enter in <**hh:mm:ss>** format. (Required)
- **Frequency**   See Section 12.3.2.2, below.

6   Enter comments concerning the resource reservation request in the **Comment** field.

7   After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save data.
  - The resource reservation must be "saved" prior to validating or rejecting. After the request has been saved, it can then be **validated** or **rejected**.
  - The selections **Validated** and **Rejected** are further discussed in Section 12.3.3.
- **Clear** to clear entries. Once cleared, the entries are deleted from the system.
- **Cancel** to exit screen without saving the request.

Table 12.3-3. Create a Resource Reservation - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**New…** button	**single-click**
3	**<resource identification information>**	**enter text, press tab**
4	**<resource scheduling information>**	**enter text, press tab**
5	**<duration information>**	**enter text, press tab**
6	**<comments>** (optional)	**enter text, press tab**
7	**Save** button	**single-click**

12.3.2.1 Selecting Resources...

Clicking on the **Resource…** button accesses a **Resources Selection** screen. The **Request Name** is blank and is to remain empty when creating a new resource reservation request. This screen provides a pair of lists: **Resources** and **Selected Resources**. The **Resources** list itemizes available resources. The **Selected Resources** list itemizes those resources that have been selected for incorporation into the resource reservation. The operator selects the desired resource(s) and, using the arrow buttons, moves the resource(s) from one list to the other list.

1   **Single-click** on your selections in the list and **single-click** on the desired arrow to move resources between the **Resources** and **Selected Resources** lists.

2   **Single-click** on one of the following buttons as appropriate:
- **OK** to save the selections and exit the screen.
- **Cancel** to exit the screen **without** saving changes.

12.3.2.2 Selecting Frequency

The **Frequency** option button provides the mechanism that allows the operator to specify whether the resource reservation request describes a one-time event or a recurring event. Clicking on **Frequency** allows the operator to specify options for periodic resource requests; that is, to specify the frequency of occurrence of a repeating resource need. Several options for expressing the frequency are available in the **Frequency** selection list box combined with a text field that provides a qualifier (i.e., number of days) for the **Every_?_days** selection only. The frequency specified defaults to **Once** to indicate that the resource need covers the entire time period covered by 'Start Time' and 'Stop Time.' Other options are identified in Table 12.3-4. The dates generated are inserted in the **Selected Intervals** list box, described in Section 12.3.2.3, below.

Table 12.3-4. Frequency Qualifiers (1 of 2)

Frequency	Text Qualifier:	Result:
Once	--	The default. Resource reservation covering the period from the start time and stop time for the start date specified.
Daily	--	Resource reservation for every day, between the start date and end date, for the start time and end time specified.
Weekly	--	Resource reservation for every week occurring on the day specified by the start date, repeated until the end date as specified.
Every_2_weeks	--	Resource reservation occurring biweekly on the day specified by the start date, repeated until the end date as specified.
Monthly	--	Resource reservation for every month on the start day of the month, repeated until the end date as specified.
Mon_thru_Fri	--	Resource reservation for every Monday through Friday, between the start date and end date, for the start time and end time specified.

Table 12.3-4. Frequency Qualifiers (2 of 2)

Frequency	Text Qualifier:	Result:
Mon_Wed_Fri	--	Resource reservation for every Monday, Wednesday, and Friday, between the start date and end date, for the start time and end time specified.
Tues_Thurs	--	Resource reservation for every Tuesday and Thursday, between the start date and end date, for the start time and end time specified.
Every_?_days	*n*	Resource reservation for every *n* days, between the start date and end date, for the start time and end time specified.
Weekend	--	Resource reservation for every Saturday and Sunday, between the start date and end date, for the start time and end time specified.

12.3.3   Edit a Resource Reservation Request

Table 12.3-5 presents (in a condensed format) the steps required to edit a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI **single-click** on the resource reservation request to be modified.

3 **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit/Definition** GUI.

4 Make the modifications to affected fields. (See Section 12.3.2, above.)
- **Interval...** is applicable when editing a resource reservation request if certain intervals are to be excluded from the resource reservation. See Section 12.3.3.1, below.

5 If appropriate at this time, **single-click** on either **Validated** or **Rejected**.
- **Validated** indicates that the reservation request is complete and 'makes sense'; that is, the request includes the appropriate resources consistent with the type of activity that is being proposed.
- **Rejected** indicates that the reservation request is rejected.
- At this time, the **Comment** field may also be updated.
- The **Status** field contains the status of the reservation request.
  - Status is system-generated based on operator-input in other fields.

6   After data is entered, **single-click** on the appropriate button(s):
- **Save** to save data and exit screen.
- **Clear** to clear entries. Once cleared, the entries are deleted from the system.
- **Cancel** to exit screen.

Table 12.3-5. Edit a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be modified)	**single-click**
3	**Modify…** button	**single-click**
4	Make modifications to affected fields	Use procedure in Section 12.3.2
5	**Validated** button or **Rejected** button if applicable	**single-click**
6	**Save** button	**single-click**

12.3.3.1   Deselecting Intervals...

The **Interval…** button provides the mechanism to tailor a **Frequency-based** request by overriding selected intervals (**Note**: the initial resource reservation must be saved prior to tailoring frequency-based requests.).   Selecting the **Interval…** button, displays a secondary screen that provides a pair of lists: **Unselected Intervals** and **Selected Intervals**. **Unselected Intervals** lists the dates that will not be reserved for the reservation request. **Selected Intervals** lists the dates that will be included for the request. The Selected Interval dates are automatically generated by the system, based upon the **Frequency** option selected (see Section 12.3.2.2, above). You can move them to or from the **Unselected Intervals** list to modify the automated list. Dates are moved from one list to the other by selecting the dates and using the arrow keys. The **Request Name** is also displayed.

1   **Single-click** on your selections and **single-click** on the desired arrow to move dates between the **Selected Intervals** and **Unselected Intervals** lists.

2   **Single-click** on one of the following buttons as appropriate:
- **OK** to save the selections and exit the screen.
- **Cancel** to exit the screen **without** saving changes.

12.3.4   Validate or Reject a Resource Reservation Request

All resource reservation requests must be validated and approved before scheduling. Validation is the process whereby a request is checked for completeness, and its purpose is deemed reasonable. After reviewing a resource reservation request, the Resource Planner may choose to consult with appropriate DAAC staff or assign a staff member ("sponsor") to validate a request. When the request is rejected, the status of the request is changed to "rejected" on the screen.

Table 12.3-6 presents (in a condensed format) the steps required to validate or reject a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
        • The **Resource Scheduler** GUI is displayed.

2   From the **Resource Scheduler** GUI **single-click** on the resource reservation request to be modified.

3   **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit** GUI.

4   **Single-click** on either **Validated** or **Rejected**.
        • **Validated** indicates that the reservation request is complete and 'makes sense'; that is, the request includes the appropriate resources consistent with the type of activity that is being proposed.
        • **Rejected** indicates that the reservation request is rejected.
        • At this time, the **Comment** field may also be updated.
        • The **Status** field contains the status of the reservation request.
            − Status is system-generated based on operator-input in other fields.

5   After data is entered, **single-click** on the appropriate button(s):
        • **Save** to save data.
        • **Clear** to clear entries. Once cleared, the entries are deleted from the system.
        • **Cancel** to exit screen.

*Table 12.3-6. Validate or Reject a Resource Reservation Request -
Quick-Step Procedures*

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be modified)	**single-click**
3	**Modify…** button	**single-click**
4	**Validated** button or **Rejected** button as appropriate	**single-click**
5	**Save** button	**single-click**

12.3.5 Approve a Resource Reservation Request

The **Approve** button is used when all reviews that are a part of the resource planning process have taken place and there are no objections to the resource usage as described by the request. Clicking on this button will verify that there are no conflicts between this resource reservation and other reservations. If conflicts are detected, a screen will pop up listing the conflicts to be addressed for resolution. Click **OK** to collapse the pop-up screen. Clicking on **Approve** generates the pop-up screen again (if conflicts exist). Approval occurs after a request has been validated and the event time is acceptable.

Table 12.3-7 presents (in a condensed format) the steps required to approve a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
   • The **Resource Scheduler** GUI is displayed.

2   From the **Resource Scheduler** GUI **single-click** on the resource reservation request to be approved.

3   **Single-click** on the **Approve** button.
   • If there are resource conflicts resulting from the attempt to approve the resource reservation request, a pop-up dialogue box appears indicating that the approval failed and making reference to the **Message Handler** GUI for further information.

4   **Single-click** on the **OK** button to collapse the pop-up dialogue box.
   • If there are no resource conflicts to be resolved, the entry in the Status column of the **Resource Scheduler** GUI indicates that the request is "Approved" (changes from "Validated"). [End of procedure.]
   • If there are resource conflicts to be resolved, the entry in the Status column of the **Resource Scheduler** GUI indicates that the request has "Conflicts" (changes from "Validated"). [Continue with Step 5.]

5   If there are resource conflicts to be resolved, examine the information displayed on the **Resource Scheduler** GUI.
   • Although the pop-up dialogue box makes reference to the **Message Handler** GUI for further information, no relevant data seems to be displayed there. Therefore, it is more appropriate to check for conflicts in the duration and frequency information for the resource reservation requests displayed on the **Resource Scheduler** GUI. When more than one resource reservation request is scheduled for the same date and time, there may be a conflict (if the same resource is specified in the requests).
   • It may be necessary to examine individual resource reservation requests in detail. If so, use the procedure to **Edit a Resource Reservation Request** (Section 12.3.3).

6   If necessary, consult with the resource requester(s), Resource Manager and other personnel to determine which resource reservation request(s) to modify or delete in order to create a conflict-free resource plan.

7   If applicable, go to the procedure to **Delete a Resource Reservation Request** (Section 12.3.8) and delete resource reservation request(s) as necessary to resolve the conflicts.

8   If applicable, go to the procedure to **Edit a Resource Reservation Request** (Section 12.3.3) and modify/validate resource reservation request(s) as necessary to resolve the conflicts.

9   If applicable, return to Step 2 to approve a modified resource reservation request.
- The modified procedure must have been "validated." If necessary, refer to the procedure to **Validate or Reject a Resource Reservation Request** (Section 12.3.4).

Table 12.3-7. Approve a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be approved)	**single-click**
3	**Approve** button	**single-click**
4	**OK** button	**single-click**
5	If there are resource conflicts to be resolved, examine the information displayed on the **Resource Scheduler** GUI	**read text**
6	Resolve conflicts as necessary	Use procedures in Sections 12.3.8, 12.3.3, and/or 12.3.4

12.3.6   Commit Resource Reservation Requests

Clicking on the **Commit globally** button commits all approved reservation requests and makes them accessible to Production Planning. All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing. Note that resource reservations/ground events cannot take effect until they have been sent to Data Processing as part of an activated production plan. (Refer to Section 13, Production Planning, to see how production plans are created and activated.)

In Data Processing a "ground event" job for each resource reservation is sent to the specified resource(s) at the indicated start time. If a data processing job is already using the specified resource(s) at the ground event's scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job.

Table 12.3-8 presents (in a condensed format) the steps required to commit a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   From the **Resource Scheduler** GUI, **single-click** on the **Commit globally** button.
         • Status shows **Committed** for all previously **Approved** requests.

2   To view a graphical representation of the resource plan execute the procedure in Section 12.3.7.

3   To exit from the **Resource Scheduler** GUI execute the procedure in Section 12.3.9.

NOTE: Resource reservations/ground events are not sent to data processing and cannot be implemented until they have been included in a production plan. Refer to Section 13, Production Planning, for the procedure on creating (including activating) production plans.

Table 12.3-8. Commit Resource Reservation Requests - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**Commit globally** button	**single-click**
2	View a graphical representation of the resource plan if desired	Use the procedure in Section 12.3.7
3	Exit from the **Resource Scheduler** GUI if desired	Use the procedure in Section 12.3.9

12.3.7   Review the Resource Timeline

The Resource Planning utilities allow the operator to view the Resource Plan as a timeline. Table 12.3-9 presents (in a condensed format) the steps required to review the Resource Timeline. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
         • The **Resource Scheduler** GUI is displayed.

2   From the **Resource Scheduler** GUI **single-click** on the **Timeline** button.
         • The **Resource Timeline** GUI is displayed.
         • The display represents a set of resources, arranged along the left side of the screen and some period of time as indicated across the top edge of the screen.

- The use of a resource over a period of time is represented by one or more 'resource reservation' bars across the screen.
- A bar represents a time period during which a resource reservation has been planned for the resource.
    - Each bar has the name of the resource reservation and a brief description.
    - For time periods during which a reservation has not been placed against a resource, that resource is planned for use by a default activity, e.g., science processing computers will be used for science processing unless a reservation has been placed against that resource.
    - Scroll bars allow scrolling up and down through the full list of resources and left and right in time.

3   Adjust the **Resource Timeline** window size and the view of the timeline as necessary using the mouse.
- Grab a corner of the timeline window with the cursor and resize the window as desired.
- Scroll up or down through the full list of resources.
- Scroll left or right to go backward or forward in time.

4   If a different time scale (start and end dates and times) is desired, perform Steps 5 through 7; otherwise, go to Step 8.

5   Execute the following menu path:

Display → Change Time Scale

- The **plan window edit** window is displayed.

6   In the **Plan Win Start** and **Plan Win End** fields of the **plan window edit** window enter the date and time for the desired start and end times using the following format:

<DD MMM YYYY hh:mm:ss>

7   When the appropriate date and time have been entered, **single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **plan window edit** window.
- **Apply** - to accept the changes without dismissing the **plan window edit** window.
- **Cancel** - to cancel the changes and dismiss the **plan window edit** window.

8   If a different time span is desired, **single-click** and **hold** on the **Show** option button, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **1 hr, 4 hr, 8 hr, 12 hr, 24 hr, 48 hr, 4 day, 1 week, 2 week, 1 month, full scale.**

9   If no resources are displayed on the GUI or if different resources should be displayed, perform Steps 10 through 14; otherwise, go to Step 15.

10   Execute the following menu path:

Display → Change resources

- The **Resource edit** window is displayed.

11   If adding resource(s) from the **Available Resources** list to the **Viewed Resources** list, select (highlight) the resource(s) to be added, then click on the **Add** button to move the resource(s) to the **Viewed Resources** list.
- Highlighted resource(s) appear(s) on the **Viewed Resources** list.

12   If deleting resource(s) from the **Viewed Resources** list, select (highlight) the resource(s) to be removed, then click on the **Del** button to remove the resource(s) from the **Viewed Resources** list.
- Highlighted resource(s) disappear(s) from the **Viewed Resources** list.

13   If changing the order in which resources are listed in the **Viewed Resources** list, select (highlight) the resource to be moved, then **single-click** on the up or down arrow as necessary to reposition the selected resource.
- Highlighted resource changes position in the **Viewed Resources** list.

14   When the **Viewed Resources** list contains the desired set of resources, **single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **Resource edit** window.
- **Apply** - to accept the changes without dismissing the **Resource edit** window.
- **Cancel** - to cancel the changes and dismiss the **Resource edit** window.

15   If different color-coding of the timeline is desired, perform Steps 16 through 20; otherwise, go to Step 21.

16   Execute the following menu path:

Display → Change colors

- The **Color Selections** window is displayed.

17   **Single-click** on the name of one of the resource reservations to be recolored.
- The resource reservation is highlighted.

18   **Single-click** on the desired color (in the color palette) to be applied to the highlighted resource reservation.

19   Repeat Steps 17 and 18 as necessary.

20   When the appropriate color changes have been made, **single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **Color Selections** window.
- **Apply** - to accept the changes without dismissing the **Color Selections** window.
- **Cancel** - to cancel the changes and dismiss the **Color Selections** window.

21   Observe the resource reservation information displayed on the **Resource Timeline** GUI.

22   Repeat the previous steps as necessary.

23   If it becomes necessary to exit from the timeline GUI execute the following menu path:

File → Quit

Table 12.3-9. Review the Resource Timeline - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**Timeline** button	**single-click**
3	**Display → Change Time Scale**	**single-click**
4	**<plan window start date and time>**	**enter text**
5	**<plan window end date and time>**	**enter text**
6	**Ok** button	**single-click**
7	**<time span>**	**single-click**
8	**Display → Change Resources**	**single-click**
9	**<resources>** (to be viewed)	**single-click**
10	**Add** button	**single-click**
11	**<viewed resource>** (to be reordered)	**single-click**
12	**<up>** arrow or **<down>** arrow (as necessary to reorder viewed resources)	**single-click**
13	**Ok** button	**single-click**
14	**Display → Change Colors**	**single-click**
15	**<resource reservation>** (to be recolored)	**single-click**
16	**<color>** for resource reservation	**single-click**

Table 12.3-9. Review the Resource Timeline - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
17	**Ok** button	**single-click**
18	Observe the resource reservation information	**read text**
19	**File → Quit** (to quit the timeline)	**single-click**

12.3.8 Delete a Resource Reservation Request

Table12.3-10 presents (in a condensed format) the steps required to delete a resource reservation request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1　　　If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
　　　　• The **Resource Scheduler** GUI is displayed.

2　　　From the **Resource Scheduler** GUI, highlight (click on) the resource reservation request you want to delete.

3　　　Execute the following menu path:

　　　　File → Delete

　　　　• Status shows "Deleted" for the selected request. The resource reservation request is not removed from the database at this point and is available for future reporting but will have no impact on resource planning. Resource reservations are removed from the Resource reservations (PDPS) database through routine database maintenance activities.

4　　　To exit from the **Resource Scheduler** GUI execute the procedure in Section 12.3.9.

Table 12.3-10. Delete a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be deleted)	**single-click**
3	**File → Delete**	**single-click**
4	Exit from the **Resource Scheduler** GUI if desired	Use procedure in Section 12.3.9

12.3.9 Shut Down the Resource Scheduler

When resource-scheduling activities have been completed, the Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the resource planning applications. If any of the three processes remains active, it is likely to interfere with subsequent attempts to launch resource planning applications.

Shutting down the Resource Scheduler starts with the assumption that the **Resource Scheduler** GUI has been launched and is currently being displayed.

Table 12.3-11 presents (in a condensed format) the steps required to shut down resource scheduling applications. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   To exit from the **Resource Scheduler** GUI when resource planning activities have been completed execute the following menu path:

   File → Exit

   - The **Resource Scheduler** GUI disappears unless there are resource reservation requests with a status of "approved".
   - If there are any resource reservation requests with a status of "approved" listed on the **Resource Scheduler** GUI, a **Close Application** pop-up dialogue box is displayed with a message "Status of the listed reservations" and a list of the resource reservation requests with "approved" status.

2   If the **Close Application** pop-up dialogue box is displayed, **single-click** on the appropriate button from the following selections:
   - **Ok** - to quit the **Resource Scheduler** GUI and dismiss the dialogue box.
     - Selecting **Ok** effectively commits all "approved" Resource Reservations.
   - **Cancel** - to dismiss the dialogue box and return to the **Resource Scheduler** GUI.

3   After quitting the **Resource Scheduler** GUI **single-click** in the UNIX window used to start the resource scheduling applications.

4   Shut down the Message Handler, System Name Server, and Resource Model by entering:

   EcPlRpSlayAll <MODE> <application_id>

   - The **Message Handler** GUI disappears.

5   To obtain a list of active processes in the specified mode enter:

   ps -ef | grep <MODE>

   - A list of active processes in the specified mode is displayed.
   - If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

     ps -auxwww | grep <MODE>

6 Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.

- None of the following processes should be active:
    - EcPlRpRe
    - EcPlRpSi
    - EcPlRpTl
    - EcPlMsh
    - EcPlSns
    - EcPlRpRm

7 If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 <process ID1> [<process ID2>] [<process ID3>] […]

8 Repeat Steps 5 through 7 as necessary.

Table 12.3-11. Shut Down the Resource Scheduler -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**File → Exit** (to quit the **Resource Scheduler** GUI)	**single-click**
2	**OK** button	**single-click**
3	UNIX window (Planning/Management Workstation)	**single-click**
4	**EcPlRpSlayAll <MODE> <application_id>**	**enter text, press Enter**
5	**ps -ef	grep <MODE>**	**enter text, press Enter**
6	Identify the resource scheduling process(es) that has (have) not shut down	**read text**
7	**kill -15 <process ID1> [<process ID2>] [<process ID3>] […]** if necessary	**enter text, press Enter**

12.4 Tuning System Parameters

The values assigned to system parameters affect the functioning and performance of the system. When certain parameters are modified, the system operates differently. Changes to some other parameters may not appear to affect the system although there may in fact be subtle effects. In any case before system parameters are modified it is essential to understand what will happen to system functioning and performance.

Many system parameters may be subject to control by Configuration Management (CM). When making or requesting a change to system parameters, the CM process at the particular site must be followed (if applicable).

Values are assigned to Data Processing Subsystem and Planning Subsystem parameters in the following databases:

- PDPS database.
- Configuration Registry database.

The Configuration Registry Server provides a single interface (via a Sybase server) for retrieving configuration attribute-value pairs for system servers from the Configuration Registry database. When system servers are started, they access the Configuration Registry Database to obtain needed configuration parameters.

The Database Administrator has access to a Configuration Registry GUI for viewing and editing configuration data in the database. Therefore, it is necessary to coordinate with the Database Administrator when changes to configuration parameters are needed. Also, as previously mentioned, changes to configuration-controlled parameters are subject to approval through the site CM process.

Default and adjusted values assigned to system parameters vary from site to site. For guidance concerning the assignment of values to parameters included in the Configuration Registry refer to document 910-TDA-022, *Custom Code Configuration Parameters for ECS*. The document is available at http://cmdm-ldo.raytheon.com/baseline/ under "Technical Documents."

The following parameters are examples of parameters whose values may be modified to enhance system functioning or performance:

- AppLogSize [parameter applies to all servers].
  - Maximum size of the application log (ALOG) file for a particular application.
  - Recommended size varies considerably depending the nature of the application for which the file is being written.
- AppLogLevel [parameter applies to all servers].
  - Level of detail provided in the ALOG file for a particular application.
  - Acceptable values are 0, 1, 2, or 3.
  - A setting of "0" provides the most data.
- DebugLevel [parameter applies to all servers].
  - Level of detail provided in the debug log file for a particular application.
  - Normally acceptable values are 0, 1, 2, or 3.
  - A setting of "0" turns off logging; a setting of "3" provides a significant amount of data.
- DpPr_MAX_RETRIES [EcDpPrEM and EcDpPrDeletion parameter (also EcDpPrQaMonitorGUI and several Science Software Integration and Test programs)].
  - Number of retries (e.g., 30) to the Science Data Server for acquires/inserts before giving up.

- DpPr_WAIT_PERIOD [EcDpPrEM and EcDpPrDeletion parameter (also EcDpPrQaMonitorGUI and several Science Software Integration and Test programs)].
    - Time in seconds (e.g., 120) to wait between retries to the Science Data Server.
- DpPrRM_MAX_RETRIES [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter].
    - Maximum number (e.g., 100) of attempts to allocate a computer resource.
- DpPrRM_RETRY_PERIOD [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter].
    - Number of seconds (e.g., 120) between retries when trying to allocate a resource.
- DpPrMaxConcurrentDPRs [EcDpPrJobMgmt parameter].
    - Maximum allowed jobs.
    - Three integer values (e.g., 100 100 100) are assigned to DpPrMaxConcurrentDPRs; the first for routine processing; the second for on-demand processing; and the third for reprocessing jobs.
- DpPrMinConcurrentDPRs [EcDpPrJobMgmt parameter].
    - Minimum allowed jobs.
    - Three integer values (e.g., 0 0 0) are assigned to DpPrMaxConcurrentDPRs; the first for routine processing; the second for on-demand processing; and the third for reprocessing jobs.
    - Minimum number of concurrent DPRs for each job class (i.e., routine, on demand, reprocessing) NOT CURRENTLY USED.
- DpPrAutoSysMaxDPRs [EcDpPrJobMgmt parameter].
    - Total number of jobs (e.g., 100) allowed in AutoSys.
- DpPrDeleteFailedPGEJobs [EcDpPrJobMgmt parameter].
    - If TRUE, failed PGE Jobs are removed by Job Management, as necessary, when space is needed for another job that is ready to run. This is recommended to keep job management straightforward. However, this may be confusing for the operator, since they may not get a chance to see the failure if the system is busy.
    - If FALSE (the usual value), failed PGE Jobs are left in AutoSys. They must not be removed manually from AutoSys, however, since they will be removed by the Production Request Editor when a Production Request or DPR is cancelled.
- DBConnections [EcPoConnections (includes EcPlSubMgr, EcPlOdMgr, EcDpPrDeletion, EcDpPrJobMgmt and EcDpPrJobMgmtClient) parameter].
    - Number of connections needed by a particular application (e.g., 10 for EcPlOdMgr).
    - Optional parameter that specifies the number of connections to maintain in the connection pool.
    - The parameter is a list of positive integers. There must be one entry for each DbHandle in the DbHandleList.
    - Generally it should be set to the maximum number of connections that are expected to be used simultaneously in a process. If one connection per thread is used, this will be the same as the number of concurrent threads expected to

execute. When the pool is used up there is a performance penalty to allocate and deallocate connections on the fly.

- − If this parameter is not specified or is given as "NONE", it defaults to 1.
- SleepDelayForFailures [EcPlSubMgr parameter].
  - − Amount of time in seconds (e.g., 60) to wait before reprocessing failed notifications. If the specified value were less than 60, a default value of 60 seconds would be assumed.
  - − Duration of the sleep delay used by the failed notification thread in seconds.
  - − Less frequent checking can increase speed for the other threads.
- SleepDelayForTimers [EcPlSubMgr parameter].
  - − Amount of time in seconds (e.g., 60) the Subscription Manager should sleep between checking for expired timers. It should be set to the minimum amount of time a timer will be set for at this DAAC. The minimum it can be set to is 60 seconds.
  - − Duration of sleep delay used by the timer checking thread in seconds.
  - − Less frequent checking can increase speed for the other threads.
- SleepDelayForExp [EcPlOdMgr parameter].
  - − Sleep delay for expiration thread in seconds (e.g., 86400).
  - − Should be considerably greater than the sleep delay for completion threads (SleepDelayForCmp).
- SleepDelayForCmp [EcPlOdMgr parameter].
  - − Sleep delay for completion threads in seconds (e.g., 300).
  - − Should be considerably less than the sleep delay for expiration threads (SleepDelayForExp).
- SocketLimit [EcDpPrDeletion, EcDpPrJobMgmt, EcPlOdMgr, EcPlSubMgr parameter].
  - − Number of connections (e.g., 200) to a server through the Hubble Space Telescope (HST) sockets middleware.
  - − Too low a number misses connections.
  - − Too high a number may adversely affect the memory of the server's host.

NOTE: When the value assigned to a parameter has been changed and saved in the Configuration Registry, the modified value does not take effect until the affected server has been restarted. For example, if the debug level for the Subscription Manager log has been changed from "2" to "3" in the Configuration Registry, the modification does not affect the recording of data in the log until after a warm restart of the Subscription Manager (at which time the server would read the parameters in the Configuration Registry).

Table 12.4-1, below, provides an Activity Checklist table of System Tuning activities.

Table 12.4-1. Tuning System Parameters - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner/ Production Planner/ Production Monitor	Monitor the Load on Processing Resources	(P) 12.4.1	

12.4.1   Monitor the Load on Processing Resources

The Production Planner and Production Monitor should work with the Resource Planner to make optimum use of processing resources. The Resource Planner allocates the disk partitions, CPUs, and RAM available for processing among the active modes (e.g., OPS, TS1, TS2).   The Production Planner and Production Monitor monitor the load on the processing resources.

The Resource Planner assigns the bulk (typically 60% - 80%) of the processing resources to the OPS mode. The remainder of the processing assets are divided among the modes used for SSI&T and new version software checkout.

The Production Planner and Production Monitor monitor the load on the processing resources to identify whether the actual load is appropriately distributed among modes. They inform the Resource Planner of under- or over-use of resources as allocated.

When monitoring the load on the processing resources, the Production Planner and Production Monitor should take the following considerations into account:

- Disk space allocated to OPS mode is likely to be used to capacity.
- Disk space assigned to the other two modes may not fill up.
- There is no one-to-one mapping of CPU allocation with actual CPUs on the science processor.
- The operating system (OS) takes care of true CPU and RAM allocation.
  - Actual CPU usage during processing is limited by the OS.
  - If ten CPUs have been specified for a particular mode, only ten Data Processing Requests (DPRs) can be running the Execute job at a given time.
  - What is really being defined is the maximum number of DPRs that will execute at a given time.
- CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs on each science processor.
- If monitoring indicates that the processor is underused when OPS mode is at full processing capacity, the number of CPUs allocated to OPS mode could probably be increased.
- If the science processor is at full capacity when OPS mode is at full processing capacity (and the processor may be overworked) the number of CPUs allocated to OPS mode should be reduced.

- Random-access memory (RAM) is subject to the same considerations as CPUs.
  - RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor.

12.4.2 Strategies for Tuning

A scenario that demonstrates how DPRs might be processed under a particular set of conditions and some strategies for tuning the system are presented in the paragraphs that follow. The processing conditions include the following types of items:

- The total number of jobs allowed into AutoSys.
- The number of CPUs available for processing.
- Characteristics of the PGEs to be processed.

The total number of jobs allowed into AutoSys is controlled by the DpPrPgeLimits table in the PDPS database. An example of some of the types of data maintained in the DpPrPgeLimits table is shown in Table 12.4-2.

Table 12.4-2. Example of PDPS Database DpPrPgeLimits Table Contents (Selected Columns)

computerName [Virtual Computer]	pgeId	maxConcurrent [DPRs]
A	1	20
B	1	20
A	2	20
B	2	20

The scenario assumes that each of the virtual computers (i.e., A and B) listed in Table 12.4-2 has 16 CPUs. (There are 32 CPUs total.)

Relevant PGE characteristics are shown in Table 12.4-3.

Table 12.4-3. PGE Characteristics

PGE	# CPUs Used	Average Execution Time	Average Stage Time	Destage Time
1	1	5 minutes	5 minutes	5 minutes
2	1	60 minutes	5 minutes	5 minutes

Assuming that 100 DPRs of each type (i.e., PGE 1 and PGE 2 - 200 DPRs total) are ready to run and are released at once into AutoSys, the following actions occur:

- Eighty (80) DPRs enter AutoSys. The remaining 120 are queued, with their assignments already made:
    - Machine (Virtual Computer) A: 20 PGE 1s start staging; 30 PGE 1s are queued on Machine A; 20 PGE 2s start staging; 30 PGE 2s are queued on Machine A.
    - Machine (Virtual Computer) B: 20 PGE 1s start staging; 30 PGE 1s are queued on Machine B; 20 PGE 2s start staging; 30 PGE 2s are queued on Machine B.
- After about five (5) minutes, all 80 DPRs that were staging have finished staging and are ready for execution. However, only 32 CPUs are available.
- The first 32 DPRs that ask for CPUs get them and start running [sixteen (16) on Machine A and sixteen (16) on Machine B]. Forty-eight (48) DPRs are waiting.
    - Assuming that in the Registry database DpPrRM_RETRY_PERIOD is set to 120 seconds and DpPrRM_MAX_RETRIES is set to 100, the waiting DPRs keep trying every two minutes for up to 100 times each before timing out (after 200 minutes).
    - Note that in this example timing out is a real possibility.
- The quick jobs complete processing after five (5) minutes, freeing up sixteen (16) CPUs. In the current example, the sixteen (16) CPUs are subsequently occupied with about eight (8) five-minute PGEs and eight (8) 60-minute PGEs because CPUs are given randomly to whichever DPR gets back first to asking for them after waiting for the retry period (i.e., 120 seconds). Priorities are not used.
    - At first, there was a 50:50 ratio of fast:slow DPRs, now there is a 25:75 ratio of fast:slow. After another five (5) minutes, the ratio becomes 12.5:87.5 fast:slow, so 87.5 % of the CPUs are occupied by 60-minute DPRs.
- Apparently, the 60-minute DPRs tend to dominate the CPUs. After one (1) hour the first batch of sixteen (16) 60-minute PGEs vacates the CPUs to be replaced by eight (8) five-minute PGEs and eight (8) 60-minute PGEs, but the five-minute PGEs become extinguished again by the slow ones.
    - If the staging and destaging times were not the same (so the DPRs didn't have the same opportunity to hit the execution stage at the same time) the scenario would proceed differently.

Various strategies can be employed to tune the system:

- Limit the number of DPRs through the use of the DpPrPgeLimitsTable.
    - In the preceding example if the number of slow DPRs allowed into AutoSys is less than the number of CPUs, there is always a channel for the fast jobs to squeeze through.
    - The big disadvantage to this approach is that the slow jobs are also being prevented from staging.
- Increase the declared number of CPUs for the processors to more than the actual number (overallocate CPUs).
    - This approach allows more of each type of PGE into the science processors.

- The disadvantage is that it could overwhelm the science computers. However, they are kept busy.
- Create new virtual computers (assigning CPUs on the processors to them) and assign (via the DpPrPgeLimits table) PGEs to run on the new virtual computers.
  - This approach is another way to guarantee bandwidth (CPUs) to PGEs.
  - The disadvantage of this approach is that some CPUs could remain idle, not being seen by one of the virtual computers.
  - In the past, there may have also been some code problems with supporting this, but those difficulties should have been resolved.

Probably some combination of the first two of the preceding strategies is best; i.e., increase the number of declared CPUs to be more than the total number of slow jobs allowed into AutoSys, always leaving some CPUs for a channel of fast jobs. The total number of faster-moving jobs should be increased to make sure that there is always be a queue of them available to get their channel occupied.

The staging and destaging times have to be accounted for and this could change things in terms of using the DpPrPgeLimits table and the number of CPUs per processor to tune the job flow.

Also, it is important to perform regular garbage collection on all of the virtual computers. Procedures for cleaning the PDPS database and DPS disks (i.e., "garbage collection") are provided in Chapter 13, Production Planning.

12.5 Troubleshooting Resource Planning Problems

Troubleshooting is a process of identifying the source of problems on the basis of observed trouble symptoms. One common source of problems involves the reliance on messages or data from other subsystems. However, unlike many other operational areas in the system, Resource Planning does not have interfaces with many other subsystems. Consequently, problems with Resource Planning can usually be traced to either some part of the Planning Subsystem or the system infrastructure.

Table 12.5-1, below, provides an Activity Checklist for troubleshooting Resource Planning problems.

Table 12.5-1. Troubleshooting Resource Planning Problems - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner	Troubleshoot a Resource Planning Problem	(P) 12.5.1	
2	Resource Planner	Check Log Files	(P) 12.5.2	
3	Resource Planner	Check Database Connections	(P) 12.5.3	

Fault Recovery

Each request that crosses a client/server boundary is assigned a system-unique identifier referred to as an RPC ID. (RPC refers to Remote Procedure Call, the mechanism by which requests are submitted from client to server.) The RPC ID facilitates the automatic fault recovery events that occur whenever there is a client or server failure.

- As a request propagates through the system, each associated client/server exchange is assigned a unique RPC ID.
    - The RPC ID for each interaction is derived from the previous RPC ID received by the client for the request. Consequently, all RPC IDs associated with a given request have a common portion that relates the various client/server calls to one another.
    - Given the previous RPC ID, clients consistently reproduce the same RPC ID that was submitted to the server on the subsequent event.
- The concept of reproducible RPC IDs is central to the system fault recovery capability.
    - When requests are retried from client to server, they are always submitted with the same RPC ID that was used in the original submission of the request, even if either the client or server has crashed between retries.
- The RPC ID is also central to the check-pointing aspect of fault recovery.
    - As requests arrive at fault recovery-enabled servers, they are recorded in a persistent store (typically a database), tagged with the RPC ID, which identifies the request.
    - As the request is serviced, check-pointing state information may be updated in the persistent store, up to and including the completion status of the request.
    - This allows the servers to resume servicing from the last check-pointed state, particularly upon resubmission from a client.

Fault Handling

Failure events are classified according to the following three severity levels:

- Fatal error.
    - Returned when a request cannot be serviced, even with operator intervention.
    - For example, if a request is made to distribute data via ftp to a non-existent host, the request is failed with a fatal error.
- Retry error.
    - Potentially recoverable error.
    - Normally, a retry error would be returned to the client only when the server cannot recover from the error automatically.
    - A retry error may require operator assistance during recovery. For example, a tape left in a tape drive might have to be removed manually.

- Warning.
  – Provided when operations can proceed without interruption, but an unexpected circumstance was detected.
  – For example, when using the Resource Scheduler GUI, the Resource Planner would enter a new name for a resource reservation request after being notified that there was a previously existing resource reservation request with the name that had been entered.

Transient errors (such as network errors) are always retry errors.

- In general, clients and servers that experience transient retry errors first attempt to recover by retrying the operation automatically.
- One special case of this is "rebinding," which refers to the process by which a client automatically attempts to re-establish communication with a server in the event communication is disrupted.
  – The disruption may be caused by transient network failure, or by the server crashing or being brought down.
  – In any case, the client automatically attempts to reconnect to the server for a configurable period of time on a client-by-client basis.

System processes encountering an error or receiving an error from a server request can either pass the error back to a higher-level client or present it to the operator for operator intervention.

Client Crash and Restart

In general when a client crashes, the server continues to service the requests that were in process at the time of the client's crash. When a client restarts in the system, it sends a restart notification to each server with which it interacts.

- Clients notify servers that they have come up either "cold" or "warm."
- Generally, the notification temperature sent to the server matches the temperature at which the client process is restarted.

The default server behavior in response to startup notification from a client is as follows:

- Warm Notification.
  – Outstanding requests for the restarted clients remain available in the persistent store.
  – The outstanding requests may be resubmitted by the client, and are serviced to completion upon resubmission.
  – Associated resources are left allocated until the requests are completed.
- Cold Notification.
  – All outstanding requests for the restarted client are cancelled.
  – If the client resubmits any cancelled request using the same RPC ID (e.g., by pressing the Retry button from an operator GUI), it is failed with a fatal error due to the client cold startup notification.
  – Any resources associated with the cancelled requests are released and reclaimed by the system.

611-EMD-001, Rev. 02

Server Crash and Restart

When a server crashes, clients cannot continue to submit requests for processing.

- Synchronous requests in progress result in a Distributed Computing Environment (DCE) exception being thrown back to the client process, which enters a rebinding failure recovery mode (as previously mentioned).
- Attempts to submit requests while the server is down result in the client blocking until a communication timeout has been reached.
- Although DCE has been replaced by socket-based library calls (i.e., CCS Middleware), the DCE exception code is handled by the CCS Middleware.

When a server restarts, it may perform various resynchronization activities in order to recover from an unexpected termination.

- In the event of a server cold start or cold restart, the server typically cancels all outstanding requests and reclaims all associated resources.
- In general, existing request queues are retained for warm restarts and cleared for cold starts or cold restarts.

12.5.1   Troubleshoot a Resource Planning Problem

1   If it is not possible to log in to the Planning Subsystem host, ask the Operations Controller/System Administrator to verify that the host is "up."
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**

2   If the GUI (i.e., the **Resource Editor** GUI or the **Resource Scheduler**) is not displayed when the start-up script has been properly invoked, ensure that the DISPLAY variable was set properly.
- For detailed instructions refer to the applicable procedure.
  - **Launch the Resource Editor** (Section 12.2.2).
  - **Launch the Resource Scheduler** (Section 12.3.1).

3   If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in a different mode from the person using the selected Application ID, use a different Application ID.
- For detailed instructions refer to the applicable procedure.
  - **Launch the Resource Editor** (Section 12.2.2).
  - **Launch the Resource Scheduler** (Section 12.3.1).

4   If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in the same mode as the person using the selected Application ID, coordinate use of Planning applications with the other user and/or the System Administrator.

5 If an error message associated with the Resource Editor is received, refer to Table 12.5-2, Resource Editor User Messages.
 - The table is adapted from the corresponding table in 609-EMD-001, *Release 7.10 Operations Tools Manual for the EMD Project*).

6 If an error message associated with the Resource Scheduler is received, refer to Table 12.5-3, Resource Scheduler User Messages.
 - The table is adapted from the corresponding table in 609-EMD-001, *Release 7.10 Operations Tools Manual for the EMD Project*).

7 If some other type of problem is encountered, check the appropriate log file.
 - For detailed instructions refer to the **Check Log Files** procedure (Section 12.5.2).

8 If the problem cannot be identified and fixed without help within a reasonable period of time, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-2. Resource Editor User Messages (1 of 3)

Message Text	Impact	Cause and Corrective Action
A resource with this name already exists - re-enter name	Each resource name in the database must be unique.	1. Enter a different name in the **Resource Name** field. 2. **Single-click** on the **Save** button.
Activity Type is Not initialized	Without this field initialized, the "Save" operation gets rejected.	1. Shut down all Resource Planning tasks. [For detailed instructions refer to the **Shut Down Resource Definition Applications** procedure (Section 12.2.7).] 2. Notify the Database Administrator to have the PDPS database initialized (run the EcPlDbBuild script in the /usr/ecs/<MODE>/CUSTOM/utilities directory). 3. Relaunch Resource Planning applications. [For detailed instructions refer to the **Launch the Resource Editor** procedure (Section 12.2.2).] 4. Resume the operation that elicited the error message.
Block Size must be an integer number – reenter	Integer only.	1. Enter the appropriate integer (e.g., 1024) in the **Block Size** field. 2. **Single-click** on the **Save** button.
Block Size required	This is a required field.	1. Enter the appropriate integer (e.g., 1024) in the **Block Size** field. 2. **Single-click** on the **Save** button.
Error modifying computer resource	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-2. Resource Editor User Messages (2 of 3)

Message Text	Impact	Cause and Corrective Action
Error modifying computer resource comments	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Error saving computer resource	The operation failed due to an error in the database interface.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Error saving computer resource comments	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Number of cpus must be an integer number	Non-numeric data are not valid.	1. Enter the appropriate integer (e.g., 22) in the **Number of CPUs** field. 2. **Single-click** on the **Save** button.
Number of cpus required	This is a required field.	1. Enter the appropriate integer (e.g., 22) in the **Number of CPUs** field. 2. **Single-click** on the **Save** button.
Operating system required	This is a required field.	1. Enter the appropriate operating system data (e.g., IRIX 6.5.22) in the **Operating System** field. 2. **Single-click** on the **Save** button.
Partition Size must be a number - reenter	Integer only.	1. Enter the appropriate integer (e.g., 400000) in the **Partition Size** field. 2. **Single-click** on the **Save** button.
Partition Size required	This is a required field.	1. Enter the appropriate integer (e.g., 400000) in the **Partition Size** field. 2. **Single-click** on the **Save** button.
Resource is reserved - cannot modify	The Resource Scheduler GUI reserves the resource.	If possible, leave the resource alone. However, if the resource definition needs to be modified immediately, use the **Resource Scheduler** to change the status or delete the reservation. 1. If the resource definition needs to be modified immediately, first delete all resource reservations that specify the resource. [For detailed instructions refer to the **Delete a Resource Reservation Request** procedure (Section 12.3.8).] 2. Modify the resource definition. [For detailed instructions refer to the **Modify a Resource** procedure (Section 12.2.5).]

Table 12.5-2. Resource Editor User Messages (3 of 3)

Message Text	Impact	Cause and Corrective Action
Resource name required	Each resource in the database must have a unique name.	1. Enter an appropriate name in the **Resource Name** field. 2. **Single-click** on the **Save** button.
Resources loaded	The resources list has been loaded from the MSS baseline configuration.	For information only. No action is necessary.
Resources not loaded - file not found	The MSS baseline configuration file is not found in the previously designated directory.	No Longer Applicable.
Select a resource to modify from the list	The selected resource should be one of the defined resources.	Select (highlight) the resource to be modified in the **Resource Name** list displayed on the **Resource Editor**.
Strings should be selected	AutoSys definition requires the association of a string name.	1. Move string resources between the **Strings** and **Associated Strings** lists as necessary by selecting (highlighting) the string to be moved, then **single-clicking** on the right or left arrow button (as applicable) to move the string to the other list. 2. **Single-click** on the **Save** button.
Total ram must be an integer number	Integer only.	1. Enter the appropriate integer (e.g., 1000) representing the computer's total RAM (in megabytes) in the **Total RAM** field. 2. **Single-click** on the **Save** button.
Total ram required	This is a required field.	1. Enter the appropriate integer (e.g., 1000) representing the computer's total RAM (in megabytes) in the **Total RAM** field. 2. **Single-click** on the **Save** button.
Unable to lock Resource tables - cannot delete resource	The processing software uses the resource or its member resource.	Do not delete the resource definition at this time. The resource definition cannot be deleted while the resource is in use. 1. Wait until the resource has been released by Data Processing. 2. Try again to delete the resource definition. [For detailed instructions refer to the **Delete a Resource** procedure (Section 12.2.6).]
Unable to lock Resource tables - cannot modify resource	The processing software uses the resource or its member resource.	Do not modify the resource definition at this time. The resource definition cannot be modified while the resource is in use. 1. Wait until the resource has been released by Data Processing. 2. Try again to modify the resource definition. [For detailed instructions refer to the **Modify a Resource** procedure (Section 12.2.5).]

Table 12.5-3. Resource Scheduler User Messages (1 of 4)

Message Text	Impact	Cause and Corrective Action
A Reservation must be selected to delete	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. **Single-click** on the resource reservation request to be deleted. 3. Execute **File → Delete** from the **Resource Scheduler** pull-down menu.
A Reservation must be selected to modify	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. From the **Resource Scheduler single-click** on the resource reservation request to be modified. 3. **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit/Definition** GUI.
Can't insert new ResvName: <name> into database	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
can't send requestActChg to resource model for resvName: <name>	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Delete ResvName: <name> from the list	The database cannot be updated.	Delete the resource reservation request. [For detailed instructions refer to the **Delete a Resource Reservation Request** procedure (Section 12.3.8).]
Error in creating a new object for row: <row>.	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-3. Resource Scheduler User Messages (2 of 4)

Message Text	Impact	Cause and Corrective Action
Fail to modify resvName: <name>	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
I can't find <Plan Name>.	There is a problem with the resource pool.	Enter a valid Plan Name.
Must select reservation	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. From the **Resource Scheduler single-click** on the desired resource reservation request.
New Resvation can't leave resources list of ResvName: <name> empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Resources** list on the **Resources Selection** GUI. [For detailed instructions refer to the **Selecting Resources...** procedure (Section 12.3.2.1).]
Open one Reservation at a time, Please	Reservation cannot be opened.	1. Find and **single-click** on the open **Resource Reservation Edit/Definition** GUI that is already open. 2. **Single-click** on either the **Save** or **Cancel** button (as appropriate). 3. Open the desired resource reservation request. [For detailed instructions refer to either the **Create a Resource Reservation Request** procedure (Section 12.3.2) or the **Edit a Resource Reservation Request** procedure (Section 12.3.3).]
PlRpSiScheduler::modifyReservation – can't save new info for resvName: <name>.	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
ResvName: < > already has status <status>.	Action cannot be completed.	Choose a different action.
ResvName: <name> can't replace new Interval List	The database cannot be updated.	1. Type a new (unique) name for the resource request in the **Request Name** field (**Resource Reservation Request Edit/Definition** GUI). 2. Click on the **Save** button. [For detailed instructions refer to r the **Create a Resource Reservation Request** procedure (Section 12.3.2).]

Table 12.5-3. Resource Scheduler User Messages (3 of 4)

Message Text	Impact	Cause and Corrective Action
ResvName: <name> Selected Intervals list can't be empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Intervals** list on the **Intervals Selection** GUI. [For detailed instructions refer to the **Deselecting Intervals...** procedure (Section 12.3.3.1).]
ResvName: <name> Selected Resources list can't be empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Resources** list on the **Resources Selection** GUI. [For detailed instructions refer to the **Selecting Resources...** procedure (Section 12.3.2.1).]
ResvName: <name> accepts new resources list	Informational message.	For information only. No action is necessary.
resvName: <name> can't replace new Resource List	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
ResvName: <name> can't uncommitted < > RActAlls	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
resvName: <name> fails to approve - status is changed to <status>.	The plan cannot be approved due to conflicts with other reservations.	Resolve the conflict(s). [For detailed instructions refer to the **Approve a Resource Reservation Request** procedure (Section 12.3.5).]
resvName: <name> myTime: <time> resourceName: <name> conflicted Time: <time> conflictResvName: <name>	Informational message.	For information only. No action is necessary.
ResvName: <name> resource list is less now	Informational message.	For information only. No action is necessary.
ResvName: <name> status is changed from approved to committed	Informational message.	For information only. No action is necessary.
ResvName: <name> status is changed to <status>	Informational message.	For information only. No action is necessary.

Table 12.5-3. Resource Scheduler User Messages (4 of 4)

Message Text	Impact	Cause and Corrective Action
Success to approve reservation Name: <name>	Informational message.	For information only. No action is necessary.
Success to update resvName: <name> name	Informational message.	For information only. No action is necessary.
This Name: <name> with status: <status> has been used, Please pick another Name.	The resource reservation request name must be unique.	1. Enter a different name in the **Request Name** field. 2. **Single-click** on the **Save** button.

12.5.2   Check Log Files

Log files can provide indications of the following types of problems:

- Communication problems.
- Database problems.
- Lack of disk space.

Table 12.5-4 presents (in a condensed format) the steps required to check log files. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the appropriate host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 12.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- **<MODE>** is current mode of operation.
  - − TS1   - Science Software Integration and Test (SSI&T)
  - − TS2   - New Version Checkout
  - − OPS   - Normal Operations
- "logs" is the directory containing resource planning log files (e.g., EcPlRpRe.ALOG, EcPlRpReDebug.log, EcPlRpSi.ALOG, or EcPlRpSiDebug.log).

3   At the command line prompt enter:

pg <file name>

- **<file name>** refers to the resource planning log file to be reviewed (e.g., EcPlRpRe.ALOG, EcPlRpReDebug.log, EcPlRpSi.ALOG, or EcPlRpSiDebug.log.
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4   Review the log file to identify problems that have occurred.
- To exit from **pg** at the **:** prompt enter:

  q
  − The command line prompt is displayed.

5   Respond to problems as follows:
- Resource Planning-related problems.
  − Perform the appropriate procedure(s) from Table 12.5-1, Troubleshooting Resource Planning Problems.
- Communication problems.
  − Notify the Operations Controller/System Administrator of suspected communication problems.
- Database problems.
  − Verify that relevant database servers are running.
  − Check for lack of (or corruption of) data in the database using either a database browser or interactive structured query language (isql) commands.
  − Notify the Database Administrator of suspected database problems.
- Lack of disk space.
  − Remove unnecessary files.
  − Notify the Operations Controller/System Administrator of recurring disk space problems.

Table 12.5-4. Check Log Files - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	identify problems indicated in the log file	**read text**
5	Respond to problems as necessary	

12.5.3 Check Database Connections

If applications (including the GUIs) are unable to connect to the database, data cannot be retrieved or (in the case of the GUIs) displayed. Consequently, if a GUI does not display data or if the display does not refresh, checking the database connections is a logical step in trying to isolate the problem.

Table 12.5-5 presents (in a condensed format) the steps required to check database connections. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Submit a request to the Database Administrator to identify the values for parameters associated with the appropriate application.
- The following parameters should be requested:
  - **DBName.**
  - **DBServer.**
  - **DBMaxConnections.**
- The preceding parameters are associated with the following applications:
  - EcPlRpRe.
  - EcPlRpRm.
  - EcPlRpSi.
  - EcPlRpTl.

2   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 12.2.1).

3   At the command line prompt enter:

isql –U <user ID> -S <database server>

- **<user ID>** is the database user's identification; e.g., **pdps_role**.
- **<database server>** is the database server; e.g., **g0sps06_srvr**.

4   At the **Password:** prompt enter:

<database password>

- **<database password>** is the password for logging in to the database using the specified **<user ID>**.
- A **1>** prompt is displayed, indicating that a connection has been made with the database.

5   At the **1>** prompt enter:

sp_who

6   At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns:
    - **spid.**
    - **status.**
    - **loginame.**
    - **hostname.**
    - **blk.**
    - **dbname.**
    - **cmd.**

7   At the **1>** prompt enter:

sp_configure "user connections"

8   At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes the following types of data:
    - **Parameter Name** (i.e., number of user connections).
    - **Default.**
    - **Memory Used.**
    - **Config Value.**
    - **Run Value.**

9   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

10   Compare the number of actual connections (results of **sp_who**) with the number of connections for which the database has been configured (results of **sp_configure "user connections"**).

11   If the number of actual connections is very close to the number of connections for which the database has been configured, notify the Database Administrator of the fact.

12   If the number of actual connections is **not** very close to the number of connections for which the database has been configured, compare the number of actual connections with the value for DBMaxConnections that the Database Administrator specified (Step 1).

13   If the number of actual connections is very close to the value for DBMaxConnections, notify the Database Administrator of the fact.

- It may be advisable to increase the value assigned to the DBMaxConnections parameter in the Configuration Registry.

Table 12.5-5. Check Database Connections - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Identify the values for database parameters associated with the appropriate Resource Planning application	**contact Database Administrator**
2	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
3	**isql –U <user ID> -S <database server>**	**enter text, press Enter**
4	**<database password>**	**enter text, press Enter**
5	**sp_who**	**enter text, press Enter**
6	**go**	**enter text, press Enter**
7	**sp_configure "user connections"**	**enter text, press Enter**
8	**go**	**enter text, press Enter**
9	**quit**	**enter text, press Enter**
10	Compare the number of actual connections with the number of connections for which the database has been configured	**read text**
11	Notify the Database Administrator of the results	**contact Database Administrator**

This page intentionally left blank.

13. Production Planning

13.1 Production Planning Process

The Planning Subsystem supports site operations in developing a production plan based on locally defined strategy. Production Planning involves creation of Production Requests using the Production Request Editor and the scheduling of the Production Requests using the Planning Workbench.

The Production Planner defines science data processing requirements in terms of Production Requests (PRs).

- A PR is an order for data to be produced by the Data Processing Subsystem.
- Each PR identifies specific science software [in the form of a Product Generation Executive (PGE)] that is to be run and the data (in terms of a time period, set of orbits, etc.) that is to be processed to produce the desired output product(s).

In response to a PR the Planning Subsystem generates either one or a series of Data Processing Requests (DPRs).

- Each DPR corresponds to one execution of a single PGE.
- Each DPR contains the information that is needed by the Data Processing Subsystem to execute the PGE and produce the desired output product(s).

The Production Planner defines Production Strategies that specify values in several categories related to DPRs.

- The values specified in a Production Strategy can be used in determining the priority of each DPR.

The Production Planner creates a Production Plan that specifies which PR(s) [and consequently which DPR(s)] should be sent to Data Processing at one time.

- When creating a Production Plan, the Production Planner can associate a particular Production Strategy with the PRs in the plan.
- When the Production Planner "activates" a Production Plan, the associated DPRs become accessible to Data Processing.

The Production Planner uses the following principal tools in the Planning Subsystem:

- Production Request Editor GUI - for creating Production Requests and Data Processing Requests.
- Production Strategies GUI - for creating Production Strategies.
- Planning Workbench GUI – for creating and activating Production Plans.
- Planning Timeline – for reviewing Production Plans.

Subsequent sections related to Production Planning address the following topics:

- Section 13.2   An overview of the process for creating or deleting Production Requests and Data Processing Requests and step-by-step procedures for using the Production Request Editor.
- Section 13.3   An overview of the process for creating, modifying, or deleting Production Strategies and step-by-step procedures for using the Production Strategies GUI.
- Section 13.4   An overview of the process for creating, activating, or deleting Production Plans and step-by-step procedures for using the Planning Workbench GUI and Planning Timeline.
- Section 13.5   An overview and step-by-step procedures for resetting/cleaning the PDPS database.
- Section 13.6   Referral to the procedures for tuning system parameters included in Chapter 14, Production Processing.
- Section 13.7   An overview of the process and step-by-step procedures for troubleshooting Production Planning problems.

Data Preprocessing (DPREP)

DPREP (data preprocessing) consists of sets of PGEs that use a statistical approach to convert Level 0 (L0) ephemeris and attitude ancillary data for a particular satellite (e.g., Terra, Aqua, or Aura) into SDP Toolkit native binary format and HDF format without altering or modifying the scientific content of the granules.

Ephemeris data provide the following types of information:

- Spacecraft location: ephemeris (or orbit) data include: latitude, longitude, and height.

Attitude data provide the following types of information:

- Orientation of the satellite, including yaw, pitch, and roll angles; and angular rates about the three axes.

DPREP PGEs are supplied by the EMD Project, unlike most PGEs, which are provided by the Science Computing Facilities that EMD supports. Release 7 DPREP supports Terra, Aqua, and Aura operations.

Terra DPREP

Terra DPREP consists of the following three PGEs:

- EcDpPrAm1EdosEphAttDPREP_PGE.
- EcDpPrAm1FddAttitudeDPREP_PGE.
- EcDpPrAm1FddEphemerisDPREP_PGE.

Operationally, EcDpPrAm1EdosEphAttDPREP_PGE and EcDpPrAm1FddAttitudeDPREP_PGE are scheduled daily and run independently of one another. EcDpPrAm1FddEphemerisDPREP_PGE is scheduled and run on an as-needed basis.

The first PGE (EcDpPrAm1EdosEphAttDPREP_PGE) produces both *ephemeris* and *attitude* data. However, the *attitude* data from the first PGE is not sure to be of good enough quality for

science data processing. The second PGE (EcDpPrAm1FddAttitudeDPREP_PGE) processes *attitude* data from FDD that is of good enough quality for science data processing. If the data quality analysis performed by the first PGE indicates that the quality of the *ephemeris* from the spacecraft is poor, DPREP initiates (by submitting a subscription) the request for replacement *ephemeris* data from FDD. The replacement ephemeris is processed by the third PGE, EcDpPrAm1FddEphemerisDPREP_PGE.

There are several sources of information on the Terra DPREP PGEs and how to run them:

- 500-EMD-001, *Terra Spacecraft Ephemeris and Attitude Data Preprocessing*.
- 611-EMD-001, *Release 7.10 Mission Operation Procedures for the EMD Project*, Chapter 26.
- Two files installed on the science processor hosts (e.g., e0spg11, or l0spg11) in the /usr/ecs/*MODE*/CUSTOM/data/DPS directory.
  - "AM1_DPREP_README"
  - "HowToRunAm1DPREP"

The Terra DPREP PGEs process Level 0 Terra (AM-1) spacecraft data (e.g., ESDT AM1ANC) provided by EDOS. In addition, the Terra DPREP PGEs use Terra FDD Attitude (AM1ATTF) data and may use Terra FDD Ephemeris (AM1EPHF) data. The output files/granules of the DPREP PGEs are subsequently used in the processing of science data from various instruments on the satellite. DPREP output granules provide valuable data concerning the location and orientation of the satellite when the data from the satellite's instruments were collected.

DPREP processing is granule-oriented; the processing interval selects data granules from the archive for DPREP to process. Then the granules get processed to completion. All data (i.e., ephemeris and attitude data) that are processed by Terra DPREP arrive in two-hour segments. Therefore, processing intervals are selected in multiples of two hours.

QA analysis is an important function of DPREP. It includes checking the continuity of the ephemeris and attitude data streams across data segments; i.e., the segment that is being processed and the *immediately* preceding and following segments. DPREP does this by performing consistency, limit, and data gap checks that bridge segment boundaries. The checks are initiated using data from the end of the *immediately* preceding segment and completed using data from the *immediately* following segment. When performing QA analysis on records close to a granule boundary, the QA window extends into the preceding or following granule as circumstances dictate. Consequently, Terra DPREP needs nominal access to the granules that immediately precede and follow the "current" granule.

Terra Ephemeris Processing

EcDpPrAm1EdosEphAttDPREP_PGE consists of a script that coordinates the following three DPREP executables:

- EcDpPrAm1EdosAncillary.
- EcDpPrAm1EdosEphemerisRepair.
- EcDpPrAm1ToolkitToHdf.

EDOS L0 Ancillary processing must run at least two hours behind "real time." As previously mentioned DPREP performs consistency checks across granule boundaries and requires the L0 ancillary granule that follows the "current" L0 ancillary granule in order to perform the consistency check. So EcDpPrAm1EdosAncillary reads in the "current" EDOS L0 Ancillary granule (ESDT AM1ANC) and the next AM1ANC granule. EcDpPrAm1EdosAncillary also reads in ephemeris and attitude data (ESDT AM1EPHN0 and ESDT AM1ATTN0) from the preceding run of the PGE to support QA analysis.

EcDpPrAm1EdosAncillary identifies data to be repaired in the ephemeris data stream. EcDpPrAm1EdosAncillary writes the ephemeris granules as temporary granules, not as Production Data Sets. If EcDpPrAm1EdosAncillary determines that the ephemeris data stream had repairable data quality problems, EcDpPrAm1EdosEphemerisRepair performs data repair on the temporary Toolkit-format ephemeris granule to produce the repaired granule. EcDpPrAm1ToolkitToHdf completes the process by generating an HDF-format EDOS ephemeris from the repaired Toolkit-format EDOS ephemeris granule. The granules that EcDpPrAm1EdosEphemerisRepair and EcDpPrAm1ToolkitToHdf produce are written as Production Data Sets. Temporary granules do not get archived, but do remain in the run-time directory for a short period of time in case they need to be examined.

EcDpPrAm1EdosEphAttDPREP_PGE produces Toolkit- and HDF-format attitude (ESDTs AM1ATTN0 and AM1ATTH0) and ephemeris (ESDTs AM1EPHN0 and AM1EPHH0) granules.

NOTE: EDOS-supplied ephemeris data are the primary source of ephemeris for Terra. However, EDOS-supplied attitude data are not the primary source of attitude for Terra. Attitude data supplied by the Flight Dynamics Division (FDD) are considered the primary source of Terra attitude.

EcDpPrAm1EdosAncillary performs a full complement of data quality analyses on the EDOS ephemeris data. In contrast EDOS-supplied attitude data are subject to minimal quality checks and never undergo data repair because EDOS-supplied attitude data are not considered good enough for science data processing. EDOS attitude data are "use at own risk" data; the data recommended for science data processing are the FDD attitude data routinely preprocessed by EcDpPrAm1FddAttitudeDPREP_PGE.

Terra FDD Attitude Processing

FDD attitude processing must run at least two hours behind "real time." DPREP performs consistency checks across granule boundaries and requires the FDD Attitude granule that follows the "current" FDD Attitude granule to perform the consistency check. Accordingly, EcDpPrAm1FddAttitudeDPREP_PGE reads in both the "current" FDD attitude granule (AM1ATTF) and the next FDD attitude granule. It also reads in the attitude granule (AM1ATTNF) it produced with its last run. The output of the process is a native-format (Toolkit) attitude file (AM1ATTNF) and an HDF-format attitude file (AM1ATTHF). A metadata file is produced for each attitude data file.

Because the FDD attitude data are extensively preprocessed by FDD prior to processing by DPREP, DPREP performs minimal data quality checking on the FDD Attitude data stream (primarily to catch transmission errors). DPREP does not perform data repair of FDD attitude data.

Terra Data Repair

Data repair is performed on the EDOS-supplied *ephemeris* data stream only. If EcDpPrAm1EdosEphAttDPREP_PGE finds too many missing data points in the ephemeris data (e.g., AM1EPHH0 and AM1EPHN0 granules have gaps of 58 records or more - or about 60 seconds), DPREP requests an FDD replacement granule (AM1EPHF) by submitting a subscription to the Spatial Subscription Server. When the FDD ephemeris granule has been ingested, EcDpPrAm1FddEphemerisDPREP_PGE preprocesses the FDD Ephemeris granule to produce the FDD Toolkit ephemeris replacement granule. The granule replaces a Toolkit-format EDOS ephemeris granule directly within the EDOS Toolkit ephemeris data stream. Consequently, FDD ephemeris processing does not generate an ESDT that stands apart from the Toolkit-format EDOS ephemeris ESDT, but instead produces a different "flavor" of ephemeris data within the Toolkit-format EDOS ephemeris data stream. Furthermore, in addition to the Toolkit-format (AM1EPHN0) FDD ephemeris granule EcDpPrAm1FddEphemerisDPREP_PGE produces an HDF-format (AM1EPHH0) FDD ephemeris granule.

Because FDD performs preprocessing on replacement data, DPREP performs minimal replacement ephemeris data quality checking (primarily to catch transmission errors). DPREP does not perform data repair on replacement ephemeris.

Terra DPREP Profiles

As previously mentioned DPREP processing has data requirements beyond the current data segment. Data from the preceding and following segments are used for performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available because adjacent segments may be in different granules. Consequently, four data processing profiles have been developed for each of the DPREP PGEs to accommodate the various permutations of data availability:

- Profile 1 is used when data are available from the preceding, current, and following segments.
- Profile 2 is used when data are available from the current and following segments only.
- Profile 3 is used when data are available from the preceding and current segments only.
- Profile 4 is used when data are available from the current segment only.

Profile 1 (data are available from the preceding, current, and following segments) is used for nominal DPREP operation. It is the profile of each DPREP step that is run on a routine basis. It is quite flexible in that it can often proceed even when there is no granule from the immediately preceding segment or from the following segment.

Profile 2 (no preceding data, but following data is available) is the boot-up process. It is used for initializing DPREP processing of the ephemeris and attitude data streams. After Profile 2 has been run on a data segment, Profile 1 (preceding and following data available) assumes processing responsibility on all data segments thereafter until data dropout or mission end is encountered.

Profile 3 (preceding data available, but no following data) processes the data segment that immediately precedes data dropout and, therefore, terminates processing on the ephemeris and attitude data streams.

Profile 4 is used for processing isolated data segments and is not likely to be scheduled operationally.

In the big picture of the mission, DPREP processing on the very first data segment would require running Profile 2 (boot-up). The next data segments would be processed using Profile 1 (nominal) processes. The very last data segment of the mission could be processed using Profile 3. However, given the processing flexibility of Profile 1, neither the scheduling of Profile 3 nor the scheduling of Profile 4 is envisioned operationally.

Aqua DPREP

Aqua DPREP consists of the following processes:

- EcDpPrPm1FddEphemerisDPREP_PGE - FDD Ephemeris Processing.
- EcDpPrPm1AttitudeDPREP_PGE - ECS Mission Operations Segment (EMOS) Attitude Processing.

There are several sources of information on the Aqua DPREP PGEs and how to run them:

- 500-EMD-002, *Aqua Spacecraft Ephemeris and Attitude Data Preprocessing*.
- 611-EMD-001, *Release 7.10 Mission Operation Procedures for the EMD Project*, Chapter 26.
- "HowToRunPm1DPREP" file installed on the science processor hosts (e.g., e0spg11, g0spg01, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory.

Aqua DPREP includes the processing of FDD ephemeris and EMOS attitude data:

- Ephemeris data is received from the FDD in ephemeris data files.
  - Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day.
- The Ground-Based Attitude Determination (GBAD) data that is used in the processing of attitude data is received from EMOS in "carry-out" files.
  - GBAD carry-out files (PMCOGBAD) are used in conjunction with ephemeris data to prepare Aqua attitude data.

DPREP processing is granule-oriented; the processing interval selects data granules from the archive for DPREP to process. Then the granules get processed to completion. Ephemeris granules that are processed by Aqua DPREP consist of 24-hour segments while attitude granules processed by Aqua DPREP consist of two-hour segments.

Operationally, Aqua Ephemeris and Attitude PGEs are scheduled daily and run independently of one another. However, ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing.

For both Aqua DPREP PGEs, the size of the QA window is three records. A window of three data points contains the record undergoing QA analysis and the immediately preceding and following records. When performing QA analysis on records close to a granule boundary in the ephemeris and attitude data streams, the QA window extends into the preceding or following data granule as circumstances dictate. Consequently, Aqua DPREP needs nominal access to the granules that immediately precede and follow the "current" granule.

Aqua Ephemeris Processing

Aqua ephemeris processing includes reformatting FDD ephemeris granules into Toolkit native format and HDF format. In addition, ephemeris metadata records are generated for the product granules.

The Aqua Ephemeris DPREP PGE (EcDpPrPm1FddEphemerisDPREP_PGE) processes Aqua FDD ephemeris granules (i.e., PM1EPHD). The output files/granules of EcDpPrPm1FddEphemerisDPREP_PGE provide satellite ephemeris data and are subsequently used in the processing of Aqua satellite attitude data.

Because of the preprocessing that FDD performs on ephemeris data, DPREP performs minimal data quality checking. The data quality analyses include continuity, consistency, range, and data gap checking, which are intended mainly to catch transmission errors. DPREP does not perform limit (spike) checking or data repair on FDD ephemeris. Any failure of data quality analysis (e.g., data gap or consistency problem) detected in the FDD ephemeris triggers a request for a replacement granule. In such cases a subscription for the FDD ephemeris replacement granule is submitted to the Spatial Subscription Server.

For the sake of timely FDD ephemeris processing, Aqua DPREP typically foregoes gap checking and continuity checking of the FDD ephemeris data timeline/data stream between the segment being processed and the one immediately following. Rather than wait 24 hours for the FDD ephemeris granule from the following segment to become available, DPREP omits gap checking and continuity checking at the following segment boundary (i.e., it reverts to Profile 3 processing). Consequently, DPREP can detect such a data gap or ensure continuity only when processing of the *following* segment completes (e.g., during the next day's run of EcDpPrPm1FddEphemerisDPREP_PGE).

Note that there is no need to forgo gap checking or continuity checking during FDD ephemeris **reprocessing** because all granules on the timeline should be available in the archive.

During processing EcDpPrPm1FddEphemerisDPREP_PGE reads in the "current" FDD PM1EPHD ephemeris granule, the immediately preceding ephemeris granule, and (if available) the immediately following ephemeris granule. In addition, it reads in a previous PM1EPHND preprocessed Aqua platform ephemeris granule in native format. The outputs of the process are preprocessed Aqua platform ephemeris granules in native format (PM1EPHND) and HDF format (PM1EPHHD). A metadata file is produced for each output data file.

Aqua Attitude Processing

Aqua Attitude DPREP (EcDpPrPm1AttitudeDPREP_PGE) inputs are from ESDTs PMCOGBAD (EMOS-supplied Ground-Based Attitude Determination carry-out files) and PM1EPHND (preprocessed Aqua platform ephemeris data in native format). The outputs of Aqua Attitude DPREP are PM1ATTHR (Aqua attitude data in HDF-EOS format) and PM1ATTNR (Aqua attitude data in native format).

The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals. The PMCOGBAD data is in the form of two-hour granules (12 granules per day). Each PM1EPHND (ephemeris) granule represents 24 hours of data (one granule per day).

The EMOS-supplied attitude consists of two data streams that are subject to QA analyses:

- Attitude data stream.
- Guidance, Navigation, and Control (GN&C) Status Word 2 data stream.
    - Contains the mode that the on-board attitude system was in (i.e., mode zero, attitude hold, sun hold, fine point, earth point, or sun point) when the attitude of the Aqua platform was recorded (at eight-second intervals).

Because the attitude data are preprocessed by EMOS prior to processing by DPREP, DPREP performs minimal data quality checking on the EMOS attitude data stream. As with ephemeris data processing, the attitude data quality analyses include continuity, consistency, range, and data gap checking, which are intended mainly to catch transmission errors. DPREP does not perform limit (spike) checking or data repair. Any failure of data quality analysis (with the exception of long data gaps) results in entering a subscription for a replacement granule from EMOS. The presence of a long gap in the EMOS attitude data timeline does not cause DPREP to request replacement data from EMOS. The gap is flagged as a long gap and remains as such in the EMOS attitude data timeline.

In general EMOS attitude processing requires raw input from the preceding segment in order to complete continuity checks between the preceding and current data segments on the Status Word 2 data stream that is imbedded within the EMOS-supplied attitude granules. Raw input is also required from the following data segment in order to complete continuity checks between the current and following data segments on the attitude data stream as well as the Status Word 2 data stream.

Because DPREP's data quality analysis includes checks for continuity across granule boundaries, EMOS attitude processing must lag at least two hours behind "real time." However, due to exceptional Aqua DPREP processing requirements, the lag is usually much longer. DPREP expects the EMOS-supplied attitude granule that follows the segment being processed to be available for continuity checking (hence the processing lag).

Interpretation of the attitude data depends on the value of the GN&C Status Word 2 contained in the carry-out file GBAD data. In general if the value of Status Word 2 represents "fine point" mode, the attitude is acceptable for Aqua science processing. Any other value indicates inadequate attitude accuracy for Aqua science data processing.

During processing EcDpPrPm1AttitudeDPREP_PGE reads in the previous, current, and next EMOS-supplied attitude granules (PMCOGBAD). It reads in the ephemeris (PM1EPHND) granule for the same time period (and adjacent granule if near a granule boundary). It also reads in the attitude granule (PM1ATTNR) produced during the previous run. The output of the process is a native-format attitude file (PM1ATTNR) and an HDF-format attitude file (PM1ATTHR). A metadata file is produced for each output data file.

Aqua Data Repair

Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed.

Aqua DPREP Profiles

As mentioned with regard to Terra DPREP processing Aqua DPREP data requirements extend beyond the current segment. Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available because adjacent segments may be in different granules. Consequently, four data processing profiles have been developed for each of the Aqua DPREP PGEs to accommodate the various permutations of data availability. The Aqua DPREP profiles were developed in accordance with the same principles as the corresponding Terra DPREP profiles. (For further information refer to the section on **Terra DPREP Profiles**.)

Aura DPREP

Operationally, Aura DPREP is very similar to Aqua DPREP. Aura DPREP consists of the following processes:

- EcDpPrAuraEphemerisDPREP_PGE - FDD Ephemeris Processing.
- EcDpPrAuraAttitudeDPREP_PGE - EMOS Attitude Processing.

There are several sources of information on the Aura DPREP PGEs and how to run them:

- 500-EMD-003, *Aura Spacecraft Ephemeris and Attitude Data Preprocessing*.
- 611-EMD-001, *Release 7.10 Mission Operation Procedures for the EMD Project*, Chapter 26.
- "HowToRunAuraDPREP" file installed on the science processor hosts (e.g., e0spg11, g0spg01, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory.

Aura DPREP includes the processing of FDD ephemeris and EMOS attitude data:

- Ephemeris data is received from the FDD in ephemeris data files.
    - Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day.
- The Ground-Based Attitude Determination (GBAD) data that is used in the processing of attitude data is received from EMOS in "carry-out" files.
    - GBAD carry-out files (AUCOGBAD) are used in conjunction with ephemeris data to prepare Aqua attitude data.

DPREP processing is granule-oriented; the processing interval selects data granules from the archive for DPREP to process. Then the granules get processed to completion. Ephemeris granules that are processed by Aura DPREP consist of 24-hour segments while attitude granules processed by Aura DPREP consist of two-hour segments.

Operationally, Ephemeris and Attitude PGEs are scheduled daily and run independently of one another. However, ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing.

For both Aura DPREP PGEs, the size of the QA window is three records. A window of three data points contains the record undergoing QA analysis and the immediately preceding and following records. When performing QA analysis on records close to a granule boundary in the ephemeris and attitude data streams, the QA window extends into the preceding or following data granule as circumstances dictate. Consequently, Aura DPREP needs nominal access to the granules that immediately precede and follow the "current" granule.

Aura Ephemeris Processing

Aura ephemeris processing includes reformatting FDD ephemeris granules into Toolkit native format and HDF format. In addition, ephemeris metadata records are generated for the product granules.

The Aura Ephemeris DPREP PGE (EcDpPrAuraEphemerisDPREP_PGE) processes Aura FDD ephemeris granules (i.e., AUREPHMF). The output files/granules of EcDpPrAuraEphemerisDPREP_PGE provide satellite ephemeris data and are subsequently used in the processing of Aura satellite attitude data.

Because of the preprocessing that FDD performs on ephemeris data, DPREP performs minimal data quality checking. The data quality analyses include continuity, consistency, range, and data gap checking, which are intended mainly to catch transmission errors. DPREP does not perform limit (spike) checking or data repair on FDD ephemeris. Any failure of data quality analysis (e.g., data gap or consistency problem) detected in the FDD ephemeris triggers a request for a replacement granule. In such cases a subscription for the FDD ephemeris replacement granule is submitted to the Spatial Subscription Server.

For the sake of timely FDD ephemeris processing, Aura DPREP typically foregoes gap checking and continuity checking of the FDD ephemeris data timeline/data stream between the segment being processed and the one immediately following. Rather than wait 24 hours for the FDD ephemeris granule from the following segment to become available, DPREP omits gap checking and continuity checking at the following segment boundary (i.e., it reverts to Profile 3 processing). Consequently, DPREP can detect such a data gap or ensure continuity only when processing of the *following* segment completes (e.g., during the next day's run of EcDpPrAuraEphemerisDPREP_PGE).

Note that there is no need to forgo gap checking or continuity checking during FDD ephemeris **reprocessing** because all granules on the timeline should be available in the archive.

During processing EcDpPrAuraEphemerisDPREP_PGE reads in the "current" FDD AUREPHMF ephemeris granule, the immediately preceding ephemeris granule, and (if available) the immediately following ephemeris granule. In addition, it reads in a previous AUREPHMN preprocessed Aura platform ephemeris granule in native format. The outputs of the process are preprocessed Aura platform ephemeris granules in native format (AUREPHMN) and HDF format (AUREPHMH). A metadata file is produced for each output data file.

Aura Attitude Processing

Aura Attitude DPREP (EcDpPrAuraAttitudeDPREP_PGE) inputs are from ESDTs AUCOGBAD (EMOS-supplied Ground-Based Attitude Determination carry-out files) and AUREPHMN (preprocessed Aura platform ephemeris data in native format). The outputs of Aura Attitude DPREP are AURATTH (Aura attitude data in HDF-EOS format) and AURATTN (Aura attitude data in native format).

The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals. The AUCOGBAD data is in the form of two-hour granules (12 granules per day). Each AUREPHMN granule represents 24 hours of data (one granule per day).

The EMOS-supplied attitude consists of two data streams that are subject to QA analyses:

- Attitude data stream.
- GN&C Status Word 2 data stream.

Because the attitude data are preprocessed by EMOS prior to processing by DPREP, DPREP performs minimal data quality checking on the EMOS attitude data stream. As with ephemeris data processing, the attitude data quality analyses include continuity, consistency, range, and data gap checking, which are intended mainly to catch transmission errors. DPREP does not perform limit (spike) checking or data repair. Any failure of data quality analysis (with the exception of long data gaps) results in entering a subscription for a replacement granule from EMOS. The presence of a long gap in the EMOS attitude data timeline does not cause DPREP to request replacement data from EMOS. The gap is flagged as a long gap and remains as such in the EMOS attitude data timeline.

In general EMOS attitude processing requires raw input from the preceding segment in order to complete continuity checks between the preceding and current data segments on the Status Word 2 data stream that is imbedded within the EMOS-supplied attitude granules. Raw input is also required from the following data segment in order to complete continuity checks between the current and following data segments on the attitude data stream as well as the Status Word 2 data stream.

Because DPREP's data quality analysis includes checks for continuity across granule boundaries, EMOS attitude processing must lag at least two hours behind "real time." However, due to exceptional Aura DPREP processing requirements, the lag is usually much longer. DPREP expects the EMOS-supplied attitude granule that follows the segment being processed to be available for continuity checking (hence the processing lag).

Interpretation of the attitude data depends on the value of the GN&C Status Word 2 contained in the carry-out file GBAD data. In general if the value of GN&C Status Word 2 represents "fine point," "attitude hold," or "earth point" mode, the attitude is acceptable for Aura science processing. Any other value indicates inadequate attitude accuracy for Aura science data processing.

During processing EcDpPrAuraAttitudeDPREP_PGE reads in the previous, current, and next EMOS-supplied attitude granules (AUCOGBAD). It reads in the ephemeris (AUREPHMN) granule for the same time period (and adjacent granule if near a granule boundary). It also reads in the attitude granule (AURATTN) produced during the previous run. The output of the process is a native-format attitude file (AURATTN) and an HDF-format attitude file (AURATTH). A metadata file is produced for each output data file.

Aura Data Repair

Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed.

Aura DPREP Profiles

As mentioned with regard to Terra and Aqua DPREP processing Aura DPREP data requirements extend beyond the current segment. Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available because adjacent segments may be in different granules. Consequently, four data processing profiles have been developed for each of the Aura DPREP PGEs to accommodate the various permutations of data availability. The Aura DPREP profiles were developed in accordance with the same principles as the corresponding Terra DPREP profiles. (For further information refer to the section on **Terra DPREP Profiles**.)

13.2 Creating/Deleting Production Requests and Data Processing Requests

From the Production Request Editor, the Production Planner can create new production requests, modify or delete production requests, and review or delete data processing requests. In addition, the Production Request (PR) Generator (EcPlPRGenerator) provides the Production Planner with a means of creating new production requests using a command-line interface (not a GUI).

Each procedure outlined has an **Activity Checklist** table that provides an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one - *Order* shows the order in which tasks should be accomplished.

Column two - *Role* lists the Role/Manager/Operator responsible for performing the task.

Column three - *Task* provides a brief explanation of the task.

Column four - **Section** provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - **Complete?** is used as a checklist to keep track of which task steps have been completed.

Table 13.2-1 provides an Activity Checklist for activities related to the creation of Production Requests and Data Processing Requests.

Table 13.2-1. Production Requests and Data Processing Requests -
Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Log in to System Hosts	(P) 13.2.1	
2	Production Planner	Launch the Production Request Editor	(P) 13.2.2	
3	Production Planner	Create a New Production Request Using the Production Request Editor GUI	(P) 13.2.3	
4	Production Planner	Create New Production Requests Using the Production Request Generator (Command-Line Interface)	(P) 13.2.4	
5	Production Planner	Edit/Modify a Production Request	(P) 13.2.5	
6	Production Planner	Delete a Production Request	(P) 13.2.6	
7	Production Planner	Review Data Processing Requests	(P) 13.2.7	
8	Production Planner	Delete a Data Processing Request	(P) 13.2.8	
9	Production Planner	Re-Generate Granules Affected by Loss of Files from the Archive	(P) 13.2.9	

NOTE: The procedures that follow are written under the assumption that PGEs have been previously created and are available for use with the same rule criteria that you are attempting to use.

13.2.1 Log in to System Hosts

Logging in to system hosts is accomplished from a UNIX command line prompt. Table 13.2-2 presents (in a condensed format) the steps required to log in to system hosts. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   At the UNIX command line prompt enter:

   setenv DISPLAY <client name>:0.0

   - Use either the X terminal/workstation IP address or the machine-name for the client name.

- When using secure shell, the DISPLAY variable is set just once, before logging in to remote hosts. If it were to be reset after logging in to a remote host, the security features would be compromised.

2   In the terminal window (at the command line prompt) start the log-in to the appropriate host by entering:

/tools/bin/ssh <host name>

- The **-l** option can be used with the ssh command to allow logging in to the remote host (or the local host for that matter) with a different user ID. For example, to log in to x0pls02 as user cmops enter:

/tools/bin/ssh -l cmops x0pls02

- Depending on the set-up it may or may not be necessary to include the path (i.e., /tools/bin/) with the ssh command. Using ssh alone is often adequate. For example:

ssh x0pls02

- or -

ssh -l cmops x0pls02

- Examples of Planning/Management Workstation host names include **e0pls03** and **l0pls02.**
- Examples of Science Processor host names include **e0spg11** and **l0spg11**.
- Examples of Queuing Server host names include **e0sps04** and **l0sps03**.
- Examples of Access/Process Coordinators (APC) Server host names include **e0acg11** and **l0acg02**.
- Examples of Ingest Server host names include **e0icg11** and **l0acg02**.
- Examples of Sun Consolidation External Server host names include **e0ins01** and **l0ins01**.
- Examples of Sun Consolidation Internal Server host names include **e0acs11 and l0acs03**.
- If you receive the message, "Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?" enter **yes** ("y" alone will not work).
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter** *passphrase* **for RSA key '**<*user@localhost*>**'** appears; continue with Step 3.
- If you have not previously set up a secure shell passphrase, go to Step 4.

3   If a prompt to **Enter passphrase for RSA key '**<*user@localhost*>**'** appears, enter:

<passphrase>

- If a command line prompt is displayed, log-in is complete.

- If the passphrase is unknown, press **Return/Enter**, which should cause a **<user@remotehost>'s password:** prompt to appear (after the second or third try if not after the first one), then go to Step 4.
- If the passphrase is entered improperly, a **<user@remotehost>'s password:** prompt should appear (after the second or third try if not after the first one); go to Step 4.

4   If a prompt for *<user@remotehost>***'s password:** appears, enter:

<password>

- A command line prompt is displayed.
- Log-in is complete.

Table 13.2-2. Log in to System Hosts - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**setenv DISPLAY <client name>:0.0**	**enter text, press Enter**
2	**/tools/bin/ssh <host name>** (as applicable)	**enter text, press Enter**
3	**<passphrase>** (if applicable)	**enter text, press Enter**
4	**<password>** (if applicable)	**enter text, press Enter**

13.2.2 Launch the Production Request Editor

The Production Request Editor is invoked from a UNIX command line prompt. Table 13.2-3 presents (in a condensed format) the steps required to launch the Production Request Editor. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
    - TS1   - Science Software Integration and Test (SSI&T)
    - TS2   - New Version Checkout
    - OPS   - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3   Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4   Start the Production Request Editor by entering:

EcPlPRE_IFStart <MODE>

- The **Production Request Editor** is launched.

Table 13.2-3. Launch the Production Request Editor - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set environment variables if necessary	**enter text, press Enter**
4	**EcPlPRE_IFStart <MODE>**	**enter text, press Enter**

13.2.3 Create a New Production Request Using the Production Request Editor GUI

The process of creating a new Production Request using the Production Request Editor GUI begins with the Production Planner starting the Production Request Editor GUI. The Production Planner specifies the PGE, duration, and comments for the new Production Request.

TYPES OF DATA PROCESSING: The system accommodates four general types of data processing.

Routine Processing   Pre-defined software production processing that is periodic and keyed to data arrival. For example, every day a Production Planner includes in the daily schedule a DPR for generating a particular Level 1A product from the most recent Level 0 data from the applicable satellite instrument.

Reprocessing   Typically involves using a new, improved PGE to process data that had previously been processed with an older version of the PGE.

Regeneration   A type of reprocessing that is performed for the purpose of replacing a missing or damaged product. It is necessary when an existing product has been corrupted or deleted. If the damaged or missing product is needed for shipping or as input for additional processing, it must be recreated. To the extent possible the regenerated file is created using the

	same input, the same processing parameters, and the same algorithm as the original file.
On-Demand Processing	Involves ad-hoc processing initiated by an end-user (as opposed to the Production Planner). For example, a researcher using data from the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) instrument on the Terra satellite may need a particular Level 2 product that has not yet been generated. The ASTER researcher would submit an on-demand request to have the product generated from a Level 1B product stored in the archive.

NOTE: When reprocessing, it is important to generate DPRs for one chain completely before generating any DPRs for the next chain. For example, if PGE1 produces Product1 that is input to PGE2 that produces Product2 that is input to PGE3 that produces Product3, PRs for PGE1, PGE2, and PGE3 should be created for each of those DPRs for Chain1. Then PRs for Chain2 can be created, etc.

A reprocessing DPR selects the latest version granule in the database as input. Consequently, if Product1 from Chain2 were in the database before the DPR for Chain1 PGE2 was created, the Chain1 DPR for PGE2 would select the Product1 from Chain2 as its input and Product1 from Chain1 would not be used.

PRODUCTION RULES: Production Rules provide templates for Instrument Teams to describe the relationship(s) between PGEs and their input and output data. The assumption of this documentation is that the user has knowledge of the specific production rules under which the PGE was created. Listed below is a sampling of the available production rules.

Basic Temporal	Temporal (time) range of inputs matches the temporal range of outputs.
Advanced Temporal	Temporal range of inputs is offset from the expected temporal range of inputs and outputs.
Alternate Inputs	PGE is run with different inputs based on the availability of various alternate input data sets.
Optional Inputs	PGE is run with specified optional inputs if available; otherwise, PGE is run without them.
Min/Max Granules	Minimum number of input granules needed for full data coverage and maximum number of input granules to search for may be specified. Minimum and maximum number of outputs expected from the PGE may be specified.
Optional DPRs	The only DPRs executed are those for which the non-routine key input data actually become available (i.e., are either produced in data processing or can be acquired from

the archive).

Metadata Checks	DPR is run only if input data's metadata value(s) meet(s) certain criteria.
Metadata Query	Input granule selection is based on metadata value(s).
Spatial Query/Spatial Pad	Input granule selection is based on the spatial coverage of another input (i.e., the key input). Spatial Pad involves adding area to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.
Closest Granule	DPR is generated if a required input granule within a particular time range (rather than an exact time) is available; otherwise, no DPR is generated.
Orbital Processing	Selection of input times is based on orbit information.
Multiple DPRs for Insertion Time	Allows the creation of DPRs for multiple granules with the same insertion time (affects ASTER L1B routine processing only).
Tiling	Input data is chosen on the basis of Instrument Team-defined tiles (geographic areas).

Table 13.2-4 presents (in a condensed format) the steps required to create a new production request using the Production Request Editor GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

*ASTER routine processing makes use of the **Optional DPRs Production Rule** to schedule and execute ASTER PGEs for new data that have been archived. The Production Planner specifies the **insertion time** range (i.e., the time period when the desired data were archived) as opposed to the **collection time** (when the satellite instrument gathered the data).*

2   **Single-click** on the **PR Edit** tab.
- The **PR Edit** GUI is displayed.

NOTE: If the GUI is unresponsive, always check to see if a prompt window is hidden behind the main GUI waiting for a response. Respond to the window, then continue with the Production Request Editor GUI.

NOTE: The **Help** buttons at the bottom of various the **PR Edit GUI** pages are non-functional (do not work).

3 **Single-click** and **hold** the **PR Type** option button to display a menu of types of production requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.
- Options are: **Routine, On-Demand** [not currently available for selection], and **Reprocessing.**

4 **Single-click** and **hold** the **User Type** option button to display a menu of types of users, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.
- Options are: **Operator, DAAC Manager, Scientist,** and **Researcher.**

NOTE: The **PR Name** and **Origination Date** fields will be filled automatically when the Production Request is saved at the end of the procedure. (You do not need to fill in these fields.)

5 In the **Originator** field either enter the actual name of the originator or enter:

<user name>

6 In the **Priority** field enter:

<priority>

- Enter a number in the range of one (1) to ten (10).
    - One (1) has the highest priority; ten (10) has the lowest priority.
- The **Priority** field specifies the **Production Request Editor** priority to be included in the Data Processing Request(s) that result(s) from the Production Request.
- **Production Request Editor Priority** is subsequently weighted according to the value specified in the Production Strategy selected from the Planning Workbench when a Production Plan is created that uses the Production Request.

7 **Single-click** on the **PGE...** button.
- The **PGE** GUI is displayed.

8 Select the desired PGE by **single-clicking** on the PGE.
- The desired **PGE** is highlighted.
- The PGE list is scrollable. (If there are items on the list in addition to those currently visible in the window, the additional items can be viewed by clicking on the arrows associated with the scrollbars.)

- The **Find** button provides a means of performing a keyword search of the **PGE Selection** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PGE Selection** table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

9   **Single-click** the appropriate button from the following selections:
- **OK** - to approve the selected PGE and dismiss the **PGE Selection** page.
  - The **Production Request - PR Edit** GUI is displayed.
  - The following fields are automatically filled in on the **Production Request - PR Edit** GUI: **Satellite Name**, **Instrument Name**, **PGE Name**, **PGE Version,** and **Profile ID.**
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without specifying a PGE.
  - The **Production Request** - **PR Edit** GUI is displayed.

10   **Single-click** on the **PGE Parameters...** button.
- The **PGE Parameters** GUI is displayed.
- PGE parameter data are displayed in a table that has the following columns:
  - **Parameter Name.**
  - **Logical Id.**
  - **Default Value.**
  - **Override Value.**
  - **Description.**

11   Select the parameter (if any) with the default value to be changed by **single-clicking** on the **Parameter Name** row.
- The parameter row is highlighted.
- Only the default values can be modified.
- Modify parameter values when and as directed by the customer (e.g., MODIS) only.

12   If a parameter with a default value to be changed has been selected, in the **Override Value** field enter:

       <override value>

13   If a parameter with a default value to be changed has been selected, **single-click** on the **Apply** button at the bottom of the **PGE Parameter Mappings** window.
- The new value is displayed in the **Override Value** column.

14   If any other parameter is to be changed, repeat Steps 11 through 13.

15 **Single-click** on the appropriate button from the following selections:

- **OK** - to approve the change(s) and dismiss the **PGE Parameter Mappings** window.
  - The **Production Request** - **PR Edit** GUI is displayed.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new value(s).
  - The **Production Request** - **PR Edit** GUI is displayed.

The procedures that follow involve the implementation of specific PRODUCTION RULES.

MODIS uses   1) Temporal Rules, to include Basic and Advanced Temporal specifications,
*2)   Orbit-Based Activation **and***
3)   Period/Calendar Specification,
4)   Conditional Activation,
*5)   Additional inputs **and***
6)   DataBase Query.

***ASTER** uses both Temporal Rules, Basic and Advanced Temporal specifications. In addition, ASTER routine processing makes use of the Optional DPRs Production Rule to schedule and execute ASTER PGEs for new data that have been archived. The Production Planner specifies the **insertion time** range (i.e., the time period when the desired data were archived) as opposed to the **collection time** (when the satellite instrument gathered the data).*

***MISR** has primarily "orbit" based PGEs.*

*Multiple "production rules" can be combined to complete a PR, however, **Temporal- and Orbit- based rules** cannot be combined.*

*To execute either a **Basic or Advanced Temporal Production Rule**, you must complete Steps 1 through 15, specified above in Section 13.2.3 and specify date and time information for processing (Steps 30 and 31). Then continue processing with Step 36, etc. as necessary.*

METADATA-BASED PRODUCTION RULE

16 If the PGE is subject to a metadata-based production rule and the value(s) to be checked need(s) to be changed, **single-click** on the **Metadata Checks…** button, then perform Steps 17 through 21 as applicable; otherwise go to Step 22.

- The **MetadataChecks** GUI page is displayed.
- The **MetadataChecks** GUI has an **InputDataType** window that lists the input data types for the PGE.
- In addition, the **MetadataChecks** GUI has a metadata checks (**MetaDataField-Operator-Value-Type**) window in which there is a table that lists the following information concerning each metadata check:
  - **MetaDataField.**
  - **Operator.**

       − **Value.**

       − **Type.**

- Initial values for metadata checks are entered during SSI&T; however, it is possible to modify the values using the **MetadataChecks** GUI when creating a production request.

       − Modify metadata check values when and as directed by the customer (e.g., MODIS) only.

17   If it is necessary to change any value(s) for metadata checks, select an input data type with a value to be changed by **single-clicking** on the corresponding row in the **InputDataType** window.

- The input data type row is highlighted.
- The metadata check information for the highlighted input data type is displayed in the **MetaDataField-Operator-Value-Type** window.
- The **Find** button provides a means of performing a keyword search of the **InputDataType** window.

       − A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **InputDataType** window that has the search text is highlighted.)

       − **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

18   Select (highlight) a metadata field with a comparison value to be changed by **single-clicking** on the corresponding row in the **MetaDataField-Operator-Value-Type** window.

- The metadata field row is highlighted in the **MetaDataField-Operator-Value-Type** window.
- The identity of the metadata field is displayed in the **MetaDataField** window.
- The **Find** button provides a means of performing a keyword search of the **MetaDataField-Operator-Value-Type** window.

       − A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **MetaDataField-Operator-Value-Type** window that has the search text is highlighted.)

       − **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

19   If it is necessary to change any value(s) for metadata checks, in the **Value** field enter:

<value>

20   **Single-click** on the appropriate button from the following selections:

- **OK** - to approve the new value and dismiss the **MetadataChecks** page.

       − The **Production Request - PR Edit** GUI is displayed.

− Go to Step 22.
- **Apply** - to approve the new value without dismissing the **MetadataChecks** page.
          − Go to Step 21.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new value.
          − The **Production Request - PR Edit** GUI is displayed.
          − Go to Step 22.

21   If any additional value(s) to be checked need to be changed, repeat Steps 17 through 20 as necessary.

ALTERNATE INPUTS PRODUCTION RULE
22   If the PGE is subject to the **Alternate Inputs Production Rule** and the timer settings or the order of alternate inputs need to be changed, **single-click** on the **Alternate Input Values…** button and perform Steps 23 through 28 as applicable; otherwise go to Step 29.
- The **AlternateInputValues** GUI page is displayed.
- The **AlternateInputValues** GUI has an **AlternateListName** window that lists the applicable alternate inputs.
- In addition, the **AlternateInputValues** GUI has an alternate input (**Order-DataType-LogicalID-Timer**) window in which there is a table that lists the following information concerning each alternate input:
          − **Order.**
          − **DataType.**
          − **LogicalID.**
          − **Timer.**
- The initial set-up for alternate inputs is entered during SSI&T; however, it is possible to modify the set-up using the **AlternateInputValues** GUI when creating a production request.

23   If it is necessary to change timer settings or the order of alternate inputs, first select (highlight) an alternate input to be changed by **single-clicking** on the corresponding row in the **AlternateListName** window.
- The alternate input row is highlighted.
- Information concerning the highlighted alternate input is displayed in the **Order-DataType-LogicalID-Timer** window.
- The **Find** button provides a means of performing a keyword search of the **AlternateListName** window.
          − A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **AlternateListName** window that has the search text is highlighted.)
          − **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

24   Select (highlight) an alternate input with timer settings or the order of alternate inputs to be changed by **single-clicking** on the corresponding row in the **Order-DataType-LogicalID-Timer** window.

- Alternate input row is highlighted in the **Order-DataType-LogicalID-Timer** window.
- The data type of the alternate input is displayed in the **DataType** field.

25   If it is necessary to change the order of alternate inputs, **single-click** on the up/down arrow buttons adjacent to the **Order-DataType-LogicalID-Timer** window as necessary until the highlighted alternate input has the proper order listed in the **Order** column of the window.

- If necessary, repeat Steps 24 and 25 to change the order of additional alternate inputs.

26   If the timer setting for an alternate input is to be modified, verify that the alternate input with the timer setting to be changed has been highlighted then in the **Timer** fields enter:

<hh:mm:ss>

- Another method of changing timer settings (other than typing the numbers) is to **single-click** in each of the timer fields in turn and click on the up/down buttons adjacent to the **Timer** fields until the correct time is indicated.

27   **Single-click** on the appropriate button from the following selections:

- **OK** - to approve the new alternate input setting(s) and dismiss the **AlternateInputValues** GUI.
  - The **Production Request - PR Edit** GUI is displayed.
  - Go to Step 29.
- **Apply** - to approve the new alternate input setting(s) without dismissing the **AlternateInputValues** GUI.
  - Go to Step 28.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new alternate input setting(s).
  - The P**roduction Request - PR Edit** GUI is displayed.
  - Go to Step 29.

28   If any additional alternate input setting(s) need to be changed, repeat Steps 23 through 27 as necessary.

Depending on the specific PGE chosen, either the **Collection Time** or **Insertion Time** button is automatically selected. **Collection Time** (time when the data were collected by the instrument on the satellite) is used for most PGEs. The **Insertion Time** option is available primarily for ASTER processing to allow the generation of DPRs for all data contained on an ASTER tape received from the ASTER Ground Data System (GDS). If the **Insertion Time** option is selected, the **Multiple DPRs** toggle button appears on the **PR Edit** GUI.

29   **Single-click** on either the **UTC Time** (Coordinated Universal Time) button or the **Orbit** button, depending on whether data to be processed is specified by time or orbit.
- If **UTC Time** is selected, go to Step 30.
- If **Orbit** is selected go to Step 33.

TEMPORAL PRODUCTION RULES

30   Enter the desired data start date and time in the **Begin** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

- As data are entered in each field the cursor automatically advances to the next field.
- Another method of entering date and time (other than typing the numbers) is to **single-click** in each of the date/time fields in turn and click on the up/down buttons adjacent to the date/time fields until the correct date/time is indicated.

31   Enter the desired data end date and time in the **End** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

MULTIPLE DPRS FOR INSERTION TIME PRODUCTION RULE

32   If the Multiple DPRs for Insertion Time Production Rule applies, ensure that the **Multiple DPRs** toggle button is depressed.
- If necessary, **single-click** on the **Multiple DPRs** toggle button.
- If the Multiple DPRs for Insertion Time Production Rule does not apply, go to Step 33.

ORBITAL PROCESSING PRODUCTION RULE

33   If the Orbital Processing Production Rule applies, in the **From** field enter:

<first orbit number>

34   If the Orbital Processing Production Rule applies, in the **To** field enter:

<last orbit number>

TILING PRODUCTION RULE

35   If the Tiling Production Rule applies, in the **Tile Id** field enter:

<tile ID>

CHAIN HEAD DESIGNATION

36 **Single-click** on the appropriate button from the following selections under **PGE Chain Head**:

- **Yes** - to designate the DPRs resulting from the PR as chain heads.
    - If designated as chain heads, the DPRs produce outputs that are to be used as inputs to downstream DPRs.
- **No** - to indicate that the DPRs resulting from the PR are not chain heads.

37 If the PGE specified in the PR is at the head of a chain and should be run on a particular virtual computer, **single-click** and **hold** the button adjacent to the **Computer** field to display a menu of available virtual computers, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- If a particular virtual computer is selected, each DPR will be scheduled to run on the corresponding machine.
- If no virtual computer is selected, the system will try to schedule the DPRs on the machine where the bulk of the DPR's accepted inputs (both static and dynamic inputs) are already staged.

INTERMITTENT ACTIVATION

38 If Intermittent Activation applies, in the **Skip** field enter:

<number of DPRs>

- If Intermittent Activation applies, perform Steps 39 and 40.
- If Intermittent Activation does not apply, go to Step 41.

39 If Intermittent Activation applies, in the **Keep** field enter:

<number of DPRs>

40 If the first DPR is to be skipped, **single-click** on the **SkipFirst** button.

41 If applicable, in the **Comments** field enter:

<comments>

42 Start the process of saving the production request and generating DPRs by executing the following menu path:

File → Save As

- The **File Selection** window is displayed.

43 In the **Selection** field enter:

<PR name>

44 **Single-click** on the **OK** button to save the production request.

- Eventually a **Production Request Explosion into DPRs** dialogue box is displayed.
  - It may take several minutes or even hours for the process to complete.
  - If the explosion into DPRs is successful, the production request and corresponding DPR(s) are saved in the PDPS database and the Production Request **PR Name** and **Origination Date** fields are automatically updated.
  - If the explosion into DPRs is not successful, the dialogue box contains a message to that effect and it will be necessary to troubleshoot the problem.

45 **Single-click** on the **OK** button to dismiss the **Production Request Explosion into DPRs** dialogue box.

- The dialogue box is dismissed.

46 To clear the entries on the **Production Request Editor** GUI execute the following menu path:

File → New

- Return to Step 3 to create another new PR.

47 To exit the **Production Request Editor** execute the following menu path:

File → Exit

Table 13.2-4. Create a New Production Request Using the
Production Request Editor GUI - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor** GUI	Use procedure in Section 13.2.2
2	**PR Edit** tab	**single-click**
3	**<PR type>** (**PR Type:** button)	**single-click**
4	**<user type>** (**User Type:** button)	**single-click**
5	**<originator>** (**Originator:** field)	**press Tab**
6	**<priority>** (**Priority:** field)	**press Tab**
7	**PGE...** button	**single-click**
8	**<PGE>** (from **PGE Selection** list)	**single-click**
9	**OK**	**single-click**
10	**PGE Parameters...** button (if applicable)	**single-click**
11	**<parameter>** (from **PGE Parameter Mappings** list) (if applicable)	**single-click**
12	**<override value>** (**Override Value** field) (if applicable)	**enter text**
13	Repeat Steps 11 and 12 (if applicable)	
14	**OK** button (if applicable)	**single-click**
15	**Metadata Checks…** button (if applicable)	**single-click**

Step	What to Enter or Select	Action to Take
16	**\<input data type\>** (from **InputDataType** list) (if applicable)	**single-click**
17	**\<metadata field\>** (from **MetaDataField-Operator-Value-Type** list) (if applicable)	**single-click**
18	**\<value\>** (**Value** field) (if applicable)	**enter text**
19	**OK** button (if applicable)	**single-click**
20	**Alternate Input Values…** button (if applicable)	**single-click**
21	**\<alternate input\>** (from **AlternateListName** list) (if applicable)	**single-click**
22	**\<alternate input\>** (from **Order-DataType-LogicalID-Timer** list) (if applicable)	**single-click**
23	**\<up\>** arrow or **\<down\>** arrow (as necessary to reorder alternate inputs) (if applicable)	**single-click**
24	**\<hh:mm:ss\>** (if applicable)	**enter text**
25	**OK** button (if applicable)	**single-click**
26	Either **UTC Time** button or **Orbit** button (as applicable)	**single-click**
27	**\<MM/DD/YYYY hh:mm:ss\>** (**Begin** fields) (if applicable)	**enter text, press Tab**
28	**\<MM/DD/YYYY hh:mm:ss\>** (**End** fields) (if applicable)	**enter text, press Tab**
29	**Multiple DPRs** toggle button (if applicable)	**single-click**
30	**\<first orbit number\>** (**From** field) (if applicable)	**enter text**
31	Enter **\<last orbit number\>** (**To** field) (if applicable)	**enter text**
32	**\<tile ID\>** (**Tile Id** field) (if applicable)	**enter number**
33	**Yes** button (under **PGE Chain Head**) (if applicable)	**single-click**
34	**\<Computer\>** (**Computer** field) (if applicable)	**single-click**
35	**\<number of DPRs\>** (**Skip** field) (if applicable)	**enter number**
36	**\<number of DPRs\>** (**Keep** field) (if applicable)	**enter number**
37	**SkipFirst** button (if applicable)	**single-click**
38	**\<comment\>** (**Comment** field) (if applicable)	**enter text**
39	**File → Save As**	**single-click**
40	**\<PR name\>** (**Selection** field)	**enter text**
41	**Ok** button	**single-click**
42	**Ok** button	**single-click**

13.2.4 Create New Production Requests Using the Production Request Generator (Command-Line Interface)

The process of creating new Production Requests using the Production Request (PR) Generator (EcPlPRGenerator) begins with the Production Planner preparing an input file for the PR Generator. Then the Production Planner starts the PR Generator, referencing the input file, and the PR Generator generates the appropriate PRs.

Before creating the new PRs the Production Planner must be prepared to provide the PGE ID (PgeId) and GEO ID (GEOId) values to be used in preparing the PRs. PgeId identifies a PGE. GEOId is the portion of a granule UR that identifies granule type, subtype, and database ID in the format ***type:subtype:dbID***. For example, SC:AST_ANC.001:14754 is the GEOId for a science granule of ASTER ancillary data with a database ID of 14754.

Table 13.2-5 presents (in a condensed format) the steps required to create new production requests using the PR Generator (command-line interface). If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Planning/Management Workstation host.
   - Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
   - For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   To set the application environment variables enter:

   setenv ECS_HOME /usr/ecs/

   - Application home environment is entered
   - When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3   In the terminal window, at the command line, enter:

   cd <path>

   - **<path>** is the path (e.g., **/usr/ecs/OPS/CUSTOM/data/PLS**) specified as the value for the EcPlPRGenerator's **InfoPath** configuration parameter.
     - Refer to the EcPlPRGenerator configuration parameters in the Registry database or the EcPlPRGenerator.CFG (EcPlPRGenerator.CFG.rgy) file.

4   Create an input file using an appropriate text editor (e.g., **vi**).

- For example:

x0pls01{cmops}[10]->vi PRfile

~

~

~

[…]

"PRfile" [New file]

– Many blank lines have been deleted from the example.
- The new file will specify the PgeId and GEOId values to be used in generating new Production Requests.
- Although this procedure has been written for the **vi** editor, any UNIX editor can be used to create the file.

5   Enter:

<PgeId>   <GEOId >

- For example the following entries are included in **PRfile** (the preceding example):

ACT#syn1#001   SC:AST_L1B.001:76677
BTS#syn1#001   SC:AST_L1B.001:76677

– **ACT#syn1#001** and **BTS#syn1#001** are PgeId values.
– **SC:AST_L1B.001:76677** is a GEOId.
- The following vi editor commands are useful:
    – **h** (move cursor left).
    – **j** (move cursor down).
    – **k** (move cursor up).
    – **l** (move cursor right).
    – **a** (append text).
    – **i** (insert text).
    – **r** (replace single character).
    – **x** (delete a character).
    – **dw** (delete a word).
    – **dd** (delete a line).
    – *n***dd** (delete *n* lines).
    – **u** (undo previous change).
    – **Esc** (switch to command mode).

6   Repeat Step 5 for each production request to be created.

7   If using **vi**, when all appropriate entries have been made in the file press the **Esc** key.

8   Save the file.
- For example, if using **vi**, enter:

    ZZ

    – **vi** exits and the new file is saved.
    – To exit **vi** without saving the new entries in the file enter:

        :q!

- UNIX prompt is displayed.

9   In the terminal window, at the command line, enter:

cd ../utilities

- Change directory to the directory containing the production planning startup scripts (e.g., EcPlPRE_IFStart).

10   To start the PR Generator enter:
EcPlPRGeneratorStart <MODE> <FileName>

11   When the PR Generator has completed its operation, enter:

cd ../logs

- Change directory to the logs directory.

12   In the terminal window, at the command line, enter:

pg EcPlPRGeneratorDebug.log

- The first page of the PR Generator debug log is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **view, vi**) can be used to review the log file.

13   Review the log file to determine the results of running the PR Generator.
- For example (extract from EcPlPRGeneratorDebug.log):

    *********** THE STATUS OF PRODUCTIONREQUESTS: **********

    ACT#syn1#001 SC:AST_L1B.001:76677
    **** The above enties in Input file Failed because of bad GEOID's*******

- To exit from **pg** at the **:** prompt enter:

    q

    – The command line prompt is displayed.

Table 13.2-5. Create New Production Requests Using the Production Request Generator (Command-Line Interface) - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**setenv ECS_HOME /usr/ecs/**	**enter text, press Enter**
3	**cd <path>**	**enter text, press Enter**
4	Create an input file using an appropriate text editor (e.g., **vi <file name>**)	**enter text, press Enter**
5	**<PgeId> <GEOId >**	**enter text, press Enter**
6	Repeat Step 5 (as necessary)	
7	**Esc** key (if/when appropriate)	**press**
8	Save the file (e.g., **ZZ**)	**enter text, press Enter**
9	**cd ../utilities**	**enter text, press Enter**
10	**EcPlPRGeneratorStart <MODE> <FileName>**	**enter text, press Enter**
11	**cd ../logs** (when appropriate)	**enter text, press Enter**
12	**pg EcPlPRGeneratorDebug.log**	**enter text, press Enter**
13	Review the log file to determine the results of running the PR Generator	**read text**

13.2.5 Edit/Modify a Production Request

Edits or modifications to a previously created production request result in a new production request. The new production request must be saved with a new name.

Table 13.2-6 presents (in a condensed format) the steps required to edit/modify a production request. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

2   **Single-click** on the **PR Edit** tab.
- The **PR Edit GUI** is displayed.

3   To display a list of Production Requests from which to select the PR to be opened execute the following menu path:

   File → Open

- A list of Production Requests is displayed in the **File Selection** window.

4   Select (highlight) the PR to be edited/modified by **single-clicking** on the corresponding PR name in the list of PRs.

5   **Single-click** on the **OK** button.
- The PR information appears in the **PR Edit GUI**.

NOTE: Perform only those steps of the procedure that are applicable to the changes you want to make. You do not have to go through all of the fields in the PR to successfully modify it. However, you must save the modified PR to make the changes effective.

6   To change the type of production request select the desired **PR Type** from the option button by **single-clicking** and **holding**, **moving** the mouse cursor to the desired selection (highlighting it), then **releasing** the mouse button.

7   To change the user type select the desired **User Type** from the option button by **single-clicking** and **holding**, **moving** the mouse cursor to the desired selection (highlighting it), then **releasing** the mouse button.

8   To change the originator in the **Originator** field either enter the actual name of the originator or enter:

        <user name>

9   To change the **Production Request Editor Priority** in the **Priority:** field enter:

        <priority>

10   If changing the PGE, **single-click** on the **PGE...** button.
- The **PGE** GUI is displayed.

11   If changing the PGE, select the desired PGE by **single-clicking** on the PGE.

12   If changing the PGE, **single-click** the **Ok** button.

13   To change PGE parameters first **single-click** on the **PGE Parameters...** button.
- The **PGE Parameters** GUI displays.

14   When changing PGE parameters, select a parameter to be modified by **single-clicking** on the **Parameter Name** row.
- The **parameter** row is highlighted.

- The **Find** button provides a means of performing a keyword search of the PGE Parameters table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the PGE Parameters table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

15   When changing PGE parameters, in the **Override Value** field enter:

<override value>

- Some PGEs may NOT have modifiable parameters.

16   When changing PGE parameters, **single-click** on the **Apply** button at the bottom of the **PGE Parameter Mappings** window.

17   When changing PGE parameters, **single-click** on the **Ok** button to approve the changes.
- The **Production Request Editor** GUI is displayed.

18   If the PGE is subject to a metadata-based production rule and the value(s) to be checked need(s) to be changed, **single-click** on the **Metadata Checks…** button.

19   If it is necessary to change any value(s) for metadata checks, select an input data type with a value to be changed by **single-clicking** on the corresponding row in the **InputDataType** window.

20   If it is necessary to change any value(s) for metadata checks, select (highlight) a metadata field with a comparison value to be changed by **single-clicking** on the corresponding row in the **MetaDataField-Operator-Value-Type** window.

21   If it is necessary to change any value(s) for metadata checks, in the **Value** field enter:

<value>

22   If it is necessary to change any value(s) for metadata checks, **single-click** on the appropriate button from the following selections:
- **OK** - to approve the new value and dismiss the **MetadataChecks** GUI.
- **Apply** - to approve the new value without dismissing the **MetadataChecks** GUI.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new value.

23   If the PGE is subject to the **Alternate Inputs Production Rule** and the timer settings or the order of alternate inputs need to be changed, **single-click** on the **Alternate Input Values…** button.

24   If it is necessary to change timer settings or the order of alternate inputs, first select (highlight) an alternate input to be changed by **single-clicking** on the corresponding row in the **AlternateListName** window.

25   If it is necessary to change timer settings or the order of alternate inputs, select (highlight) an alternate input with timer settings or the order of alternate inputs to be changed by **single-clicking** on the corresponding row in the **Order-DataType-LogicalID-Timer** window.

26   If it is necessary to change the order of alternate inputs, **single-click** on the up/down arrow buttons adjacent to the **Order-DataType-LogicalID-Timer** window as necessary until the highlighted alternate input has the proper order listed in the **Order** column of the window.

27   If the timer setting for an alternate input is to be modified, verify that the alternate input with the timer setting to be changed has been highlighted then in the **Timer** fields enter:

     <hh:mm:ss>

28   If it is necessary to change timer settings or the order of alternate inputs, **single-click** on the appropriate button from the following selections:
- **OK** - to approve the new alternate input setting(s) and dismiss the **AlternateInputValues** GUI.
- **Apply** - to approve the new alternate input setting(s) without dismissing the **AlternateInputValues** GUI.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new alternate input setting(s).

29   **Single-click** on either the **UTC Time** (Coordinated Universal Time) button or the **Orbit** button, depending on whether data to be processed is specified by time or orbit.

NOTE: Depending on the specific PGE chosen, either the **Collection Time** or **Insertion Time** button is automatically selected. If the **Insertion Time** option is selected, the **Multiple DPRs** toggle button appears on the **PR Edit** GUI.

30   If **UTC Time** is to be changed, enter the desired data start date and time in the **Begin** fields in the following format:

     <MM/DD/YYYY hh:mm:ss>

31   If **UTC Time** is to be changed, enter the desired data end date and time in the **End** fields in the following format:

     <MM/DD/YYYY hh:mm:ss>

32   If the Multiple DPRs for Insertion Time Production Rule applies, ensure that the **Multiple DPRs** toggle button is depressed.

  - If necessary, **single-click** on the **Multiple DPRs** toggle button.

33   If the Orbital Processing Production Rule applies and the first orbit of data is to be modified, in the **From** field enter:

  <first orbit number>

34   If the Orbital Processing Production Rule applies and the last orbit of data is to be modified, in the **To** field enter:

  <last orbit number>

35   If the Tiling Production Rule applies, in the **Tile Id** field enter:

  <tile ID>

36   **Single-click** on the appropriate button from the following selections under **PGE Chain Head**:

  - **Yes** - to designate the DPRs resulting from the PR as chain heads.
  - **No** - to indicate that the DPRs resulting from the PR are not chain heads.

37   If the PGE specified in the PR is at the head of a chain and should be run on a particular virtual computer, **single-click** and **hold** the button adjacent to the **Computer** field to display a menu of available virtual computers, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

38   If Intermittent Activation applies and the number of DPRs to skip is to be modified, in the **Skip** field enter:

  <number of DPRs>

39   If Intermittent Activation applies and the number of DPRs to keep is to be modified, in the **Keep** field enter:

  <number of DPRs>

40   If Intermittent Activation applies and the first DPR is to be skipped, verify that the **SkipFirst** button has been activated (**single-click** on the button if necessary).

41   If applicable, in the **Comments** field enter:

  <comments>

42   To save the modified production request execute the following menu path:

File → Save As

- **File Selection** Window appears.

43   In the **Selection** field enter:

<PR name>

- Production Request is named.
- Name must be changed.

44   **Single-click** on the **Ok** button.
- Eventually a **Production Request Explosion into DPRs** dialogue box is displayed.

45   **Single-click** on the **OK** button to dismiss the **Production Request Explosion into DPRs** dialogue box.
- The dialogue box is dismissed.

Table 13.2-6. Edit/Modify a Production Request - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**PR Edit** tab	**single-click**
3	**File → Open**	**single-click**
4	**<PR name>** (from **Production Requests:** list)	**single-click**
5	**OK** button	**single-click**
6	**<PR type>** (**PR Type:** button) (if applicable)	**single-click**
7	**<user type>** (**User Type:** button) (if applicable)	**single-click**
8	**<originator>** (**Originator:** field) (if applicable)	**enter text**
9	**<priority>** (**Priority:** field) (if applicable)	**enter number**
10	**PGE...** button (if applicable)	**single-click**
11	**<PGE>** (from **PGE Selection** list) (if applicable)	**single-click**
12	**OK** button (if applicable)	**single-click**
13	**PGE Parameters...** button (if applicable)	**single-click**
14	**<parameter>** (from **PGE Parameter Mappings** list) (if applicable)	**single-click**
15	**<override value>** (**Override Value** field) (if applicable)	**enter text**
16	**OK** button (if applicable)	**single-click**
17	**Metadata Checks…** button (if applicable)	**single-click**
18	**<input data type>** (from **InputDataType** list) (if applicable)	**single-click**
19	**<metadata field>** (from **MetaDataField-Operator-Value-Type** list) (if applicable)	**single-click**

Step	What to Enter or Select	Action to Take
20	**<value>** (**Value** field) (if applicable)	**enter text**
21	**OK** button (if applicable)	**single-click**
22	**Alternate Input Values…** button (if applicable)	**single-click**
23	**<alternate input>** (from **AlternateListName** list) (if applicable)	**single-click**
24	**<alternate input>** (from **Order-DataType-LogicalID-Timer** list) (if applicable)	**single-click**
25	**<up>** arrow or **<down>** arrow (as necessary to reorder alternate inputs) (if applicable)	**single-click**
26	**<hh:mm:ss>** (**Timer** fields) (if applicable)	**enter text**
27	**OK** button (if applicable)	**single-click**
28	Either **UTC Time** button or **Orbit** button (as applicable)	**single-click**
29	**<MM/DD/YYYY hh:mm:ss>** (**Begin** fields) (if applicable)	**enter text, press Tab**
30	**<MM/DD/YYYY hh:mm:ss>** (**End** fields) (if applicable)	**enter text, press Tab**
31	**Multiple DPRs** toggle button (if applicable)	**single-click**
32	**<first orbit number>** (**From** field) (if applicable)	**enter number**
33	**<last orbit number>** (**To** field) (if applicable)	**enter number**
34	**<tile ID>** (**Tile Id** field) (if applicable)	**enter number**
35	**Yes** button (under **PGE Chain Head**:) (if applicable)	**single-click**
36	**<Computer>** (from **Computer** field) (if applicable)	**single-click**
37	**<number of DPRs>** (**Skip** field) (if applicable)	**enter number**
38	**<number of DPRs>** (**Keep** field) (if applicable)	**enter number**
39	**SkipFirst** button (if applicable)	**single-click**
40	**<comment>** (**Comment** field) (if applicable)	**enter text**
41	**File → Save As**	**single-click**
42	**<PR name>** (**Selection** field)	**enter text**
43	**Ok** button	**single-click**
44	**Ok** button	**single-click**

13.2.6 Delete a Production Request

Production Requests can be deleted if necessary. Table 13.2-7 presents (in a condensed format) the steps required to delete Production Requests. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

2   **Single-click** on the **PR List** tab.
- The **PR List** GUI is presented.
- A list of Production Requests is displayed.

3   If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.
- Options are: **All** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.
- A list of Production Requests of the specified type is displayed.
- A **Refresh** button is available for refreshing the list if necessary.
- The **Find** button provides a means of performing a keyword search of the **PR List** GUI page.
  – A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PR List** GUI page that has the search text is highlighted.)
  – **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

4   **Single-click** on the Production Request to be deleted.
- The Production Request to be deleted is highlighted.

5   Execute the following menu path:

Edit→Delete

- A dialogue box is displayed requesting confirmation of the decision to delete the Production Request.

6   **Single-click** on the **OK** button to delete the Production Request.
- If there are no dependencies among Production Requests, a confirmation dialogue box is displayed.
- If there are dependencies among Production Requests, a "List Of Orphan DPRs" dialogue box is displayed.
- The **Find** button provides a means of performing a keyword search of the "List Of Orphan DPRs" dialogue box.
  – A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the "List Of Orphan DPRs" dialogue box that has the search text is highlighted.)
  – **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

7   **Single-click** on the appropriate button from the following selections:
- **OK** - to confirm deletion of the Production Request(s) and dismiss the dialogue box.
  − A "deletion completed" dialogue box is displayed.
  − The Production Request(s) is/are deleted.
- **Cancel** - to preserve the Production Request(s) and dismiss the dialogue box.
  − The dialogue box is dismissed.
  − The Production Request(s) is/are not deleted.

8   **Single-click** on **OK** to dismiss the "deletion completed" dialogue box.
- The "deletion completed" dialogue box is dismissed.

9   To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

10   **Single-click** on the appropriate button from the following selections:
- **OK** - to exit from the Production Request Editor GUI.
  − The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-7. Delete a Production Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**PR List** tab	**single-click**
3	**<PR name>** (from **Production Requests** list)	**single-click**
4	**Edit→Delete**	**single-click**
5	**OK** button	**single-click**
6	**OK** button	**single-click**
7	**OK** button	**single-click**

13.2.7 Review Data Processing Requests

The process of reviewing Data Processing Requests begins with the Production Planner launching the Production Request Editor. The Production Planner can review DPRs associated with a specific PR. The Production Planner can review such DPR values as input granule(s), output granule(s), predicted and actual start times, data start time, status, and priority.

Table 13.2-8 presents (in a condensed format) the steps required to display and review DPRs. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

2   **Single-click** on the **DPR List** tab.
- The **DPR List** GUI is presented.

3   If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.
- Options are: **ALL** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.

4   **Single-click** on the option button associated with the **PR Name** field.
- A list of Production Requests is displayed.

5   **Single-click** on the Production Request that includes the DPR to be reviewed.
- The Production Request is entered into the **PR Name** field.

6   **Single-click** on the **Filter** button.
- The DPRs for the specified PR only are shown in the **Data Processing Requests** table.

7   **Single-click** on the DPR to be reviewed from those shown in the **Data Processing Requests** table.
- The DPR to be reviewed is highlighted.
- The **Find** button provides a means of performing a keyword search of the **Data Processing Requests** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **Data Processing Requests** table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

8   Execute the following menu path:

File → Open

9   **Single-click** on the **DPR View** tab.
- The **DPR View** GUI is displayed.
- The **DPR** information is displayed on the **DPR View** GUI.

10   Review **Data Processing Request Identification** information displayed.
- **DPR Name.**
- **PR Name.**
- **Origination Date**

- **Originator**
- **PGE Id.**
- **Data Start Time.**
- **Data Stop Time.**
- **Predicted Start Time (date and time).**
- **Actual Start Time (date and time).**
- **Priority.**
- **Status.**

11   **Single-click** on the **PGE Parameters...** button.
- The **PGE Parameter Mappings** GUI displays.

12   Review the **PGE Parameter Mappings** information displayed.
- PGE parameter data are displayed in a table that has the following columns:
    – **Parameter Name.**
    – **Logical Id.**
    – **Default Value.**
    – **Override Value.**
    – **Description.**
- The **Find** button provides a means of performing a keyword search of the PGE Parameters table.
    – A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the PGE Parameters table that has the search text is highlighted.)
    – **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

13   When finished reviewing the PGE parameters, **single-click** on the **OK** button.
- The **PGE Parameter Mappings** GUI is dismissed.
- The **DPR View** GUI is displayed.

14   **Single-click** on the **PGE File Mappings...** button.
- The **UR File Mappings** GUI is displayed.

15   Review the **Universal Reference (UR) File Mappings** information displayed.
- **Input Data.**
    – **Logical Id.**
    – **Granule Id.**
    – **Start Time(UTC)** (date and time).
    – **Stop Time (UTC)** (date and time).
    – **Availability** (date and time).
    – **UR** (granule universal reference).
- **Output Data** (Displays the same data as shown for Input Data).

- The **Find** buttons provides a means of performing keyword searches of the **Input Data** and **Output Data** tables.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. [The first entry in the **Input Data** or **Output Data** table (as applicable) that has the search text is highlighted.]
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

16   When finished reviewing the **UR File Mappings**, **single-click** on the **OK** button.
- The **UR File Mappings** GUI is dismissed.
- The **DPR View** GUI is displayed.

17   Review the **DPR Status** information displayed.
- **Predicted Start Time** (date and time).
- **Actual Start Time** (date and time).
- **Priority.**
- **Status.**
- **Predicted** and **Actual Start Times** and **Status** are not displayed if the Production Request has not been scheduled.

18   Repeat Steps 2 through 17 to review multiple DPRs associated with multiple PRs.

19   To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

20   **Single-click** on the appropriate button from the following selections:
- **OK** - to exit from the **Production Request Editor** GUI.
  - The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-8. Display Data Processing Request Information - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**DPR List** tab	**single-click**
3	**<PR name>** (from **PR Name** option list)	**single-click**
4	**Filter** button	**single-click**
5	**<DPR ID>** (from **Data Processing Requests** list)	**single-click**
6	**File → Open**	**single-click**

Step	What to Enter or Select	Action to Take
7	**DPR View** tab	**single-click**
8	Review DPR information	**read text**
9	**PGE Parameters...** button	**single-click**
10	Review PGE parameters	**read text**
11	**OK** button	**single-click**
12	**PGE File Mappings...** button	**single-click**
13	Review input and output granule information	**read text**
14	**OK** button	**single-click**
15	Repeat Steps 2 through 14 to review additional DPRs	

13.2.8 Delete a Data Processing Request

DPRs can be deleted manually if necessary. Table 13.2-9 presents (in a condensed format) the steps required to delete DPRs. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

2   **Single-click** on the **DPR List** tab.
- The **DPR List** GUI is presented.

3   If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.
- Options are: **ALL** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.

4   **Single-click** on the **PR Name** option button.
- A list of Production Requests is displayed.

5   **Single-click** on the Production Request for which a DPR listing is desired.
- The Production Request is entered into the **PR Name** field.

6   **Single-click** on the **Filter** button.
- A list of the DPRs associated with the Production Request is displayed.

7   **Single-click** on the DPR to be deleted.
- The DPR to be deleted is highlighted.

- The **Find** button provides a means of performing a keyword search of the **Data Processing Requests** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **Data Processing Requests** table that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

8   Execute the following menu path:

Edit→Delete

- A DPR deletion confirmation dialogue box is displayed requesting confirmation of the decision to delete the DPR.

9   Click on the appropriate button from the following selections:
- **OK** - to confirm deletion of the DPR and dismiss the dialogue box.
  - A deletion dialogue box is displayed.
  - The DPR is deleted.
  - If the DPR was the only DPR in the Production Request, the PR is deleted too.
- **Cancel** - to preserve the DPR and dismiss the dialogue box.
  - The deletion confirmation dialogue box is dismissed.
  - The DPR is not deleted.

10   Select **OK** to dismiss the dialogue box.
- The dialogue box is dismissed.

11   To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

12   **Single-click** on the appropriate button from the following selections:
- **OK** - to exit from the **Production Request Editor** GUI.
  - The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-9. Delete a Data Processing Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**DPR List** tab	**single-click**
3	**<PR name>** (from **PR Name** option list)	**single-click**
4	**Filter** button	**single-click**
5	**<DPR ID>** (DPR to be deleted) (from **Data Processing Requests** list)	**single-click**
6	**Edit → Delete**	**single-click**
7	**OK** button	**single-click**
8	**OK** button	**single-click**

13.2.9 Re-Generate Granules Affected by Loss of Files from the Archive

The entry point for this procedure is the "Data Recovery Process" in Chapter 17, including (especially) the "Data Recovery Procedure for Known Files" and "SDSRV Retrieval of Granule Production History Metadata" sections.

The role of this procedure is to initiate production of replacements for granules previously generated within this DAAC but now lost due to failure in the Archive. In order to achieve this, Production Requests (PRs) for the generation of the replacement granules must be created, entered into a Production Plan, and activated.

The PRs are created using the Production Request Editor (PRE). However, first the necessary parameters of the PRs must be retrieved from the Production History file (PH) of the lost granule. The PH file is itself acquired by use of the QA Monitor tool and, though the PH UR is supplied in the input to this process, this can be achieved only by first using the QA Monitor tool to search the SDSRV database for granules matching the lost granule. Hence, this process commences with use in the QA Monitor GUI of the attributes of each lost granule (shortname, versionID etc.) supplied in the input to this procedure. Refer to Chapter 17 for further information on the wider context of this procedure.

The input to this procedure is the "Granules for PDPS Re-Generation" list. This is a list of granules and associated metadata generated by the procedure "SDSRV Retrieval of Granule Production History Metadata" (refer to Chapter 17). While the list can be obtained electronically, its use in this procedure is line-by-line. However, if available electronically, entries from it can be 'cut and pasted' into the input fields of the QA monitor GUI to avoid typing errors.

The outputs of the procedure are as follows:
- Granule re-generation Production Requests entered into a production plan.
- The "PDPS Residual Granules List" which is a list of Granules which this PDPS cannot re-generate or which it has been decided do not justify re-generation. This list is returned to the process in Chapter 17.

Note that the following assumptions apply to the application of this procedure:

- The Science Data Server (SDSRV) will provide a list of granules to be regenerated ("Granules for PDPS Re-Generation"). This list will contain information about the granules to be regenerated and a Universal Reference (UR) for the associated Production History tar file.
- When reproducing lost granules, all outputs of the PGE, not just those equivalent to the lost granule(s), will be produced and archived.
- It cannot be guaranteed that the PGE re-run will use the same inputs as were used during its original execution due to the variability of: Optional/Alternate inputs, Metadata Checks, Metadata Query and other Production Rules.
- It is possible that at the time of the original run of PGE, certain optional/alternate inputs were not available, which became available later. During the re-run of the PGE use of those additional or other optional inputs cannot be avoided. However, it can be assumed that an equivalent or better product than the original will be produced as a result.
- PDPS maintains a minimal amount of granule level versioning. By design, only the latest version of the granule will be used. If the PGE which is to be re-run uses inputs which have more than one granule level version, PDPS will use only the latest version of those inputs.   However, if references to those granules have been deleted from the PDPS database (a delete script, which runs periodically, cleans up unused database entries), then PDPS will choose the first one returned from SDSRV. SDSRV does not guarantee any sort of ordering in this case but PDPS will select the latest granule from those returned (Note: depends on fix to NCR ECSed16326).
- At Production Request time, the default values for metadata checks can be overridden. The new values used are stored in the PDPS database but not in the Production History. If, at the time of re-run of the PGE, the references to that PGE have been deleted from PDPS database, the default metadata checks will be used. It is possible, therefore, that these default values will cause this DPR not to be run in this instance; e.g. if the metadata checks had been changed in the original run to be less restrictive. If these types of changes to metadata checks are required in order to get DPRs to run, then it is assumed that these defaults are saved as part of the PGE profile.
- In a manner that is similar to Metadata Checks, a Metadata Query specifies a query to be used to determine the input granules to be used for a DPR. For reasons of production timing or updated QA values, a product reproduced at a later date could have different granules used as input. Again, it can be assumed that in this case a better product will result.
- Other production rules (e.g., spatial query) could make it impossible to reproduce identical granules.
- If a PGE (PGE name, version and profile) has to support lost granules regeneration, then that PGE should not be deleted from the PDPS database. This means, in the SSIT Operational Metadata GUI, the delete flag for that PGE should not be checked.

Table 13.2-10 presents (in a condensed format) the steps required to re-generate granules affected by loss of files from the archive. If you are already familiar with the procedures, you

may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Determine whether all the granules in the input list "Granules for PDPS Re-Generation" should be reproduced.
- It is possible that some granules need not be reproduced, e.g. because a newer version of that product is available.
- Granules that need not be reproduced should be added to the "PDPS Residual Granules" list. Also, at any time during this process, if it is determined that some granules cannot be regenerated or need not be regenerated, then those granules should be included in the "PDPS Residual Granules" list.

2   Retrieve (using the QA Monitor GUI) the Production History tar file from the archive for each granule in the "Granules for PDPS Re-Generation" list that needs to be reproduced.
- Procedures for using the QA Monitor GUI are presented in Chapter 15.
- Use the lost granule's datatype, and "begin date" and "end date" values that encompass its RangeBeginningDateTime and RangeEndingDateTime. Note that the GUI interprets the dates in UTC format.
- For each granule that meets the query conditions and is displayed on the QA Monitor GUI, the granule's UR, its Production History tar file's UR, and the name of the Production History tar file are shown. For only one of the granules will the URs (both the granule UR and the Production History UR) match the URs for this granule in the input list.
    − If the Query failed or did not return any hit that matched, add the granule to "PDPS Residual Granules" list.
- Retrieve the matching granule.
    − The Production History tar file will be acquired to a directory that is configurable (the name of the configuration parameter is DpPrQA_DATA_DIR, and the default value for this parameter is $ECS_HOME/<MODE>/CUSTOM/data/DPS).
    − The name of the tar file is the one that appears under the column **Production History File Name** on the QA Monitor GUI.
- Note that if more than one granule in the input list maps to the same Production History tar file, then the Production History tar file need not be retrieved multiple times.
- The Production History tar file contains the Process Control File (PCF) which has all the information needed to re-run the PGE. The following particulars have to be extracted from the PCF:
    − PGE Name.
    − PGE Version.
    − PGE Profile ID.
    − DPR Start time.
    − DPR Stop time.
    − PGE runtime parameters and their associated values.

- Identification of information in the PCF:
  - The PGE Name, PGE Version, and the PGE Profile appear in the System Runtime Parameters section of the PCF. They are concatenated (with a # sign to separate them) and appear in the place reserved for "Software ID".
  - DPR Start time appears in the User Defined Parameter Section of the PCF under the logical ID 10258.
  - DPR Stop time also appears in the User Defined Parameter Section of the PCF under logical ID 10259.
  - All other logical IDs in the User Defined Parameter Section of the PCF form the run time parameters and their associated values. Note the logical ID and its corresponding values.
  - An automated script could be written to extract the values from the PCF.

3 If the PGE name, version and profile that is extracted from the PCF does not appear as an Existing/New PGE, then add the granule that is to be regenerated to the "PDPS Residual Granules" list.

4 From the SSIT host, launch the **SSIT Manager** GUI and invoke the **PDPS Operational Metadata** GUI.
- Procedures for using SSI&T tools are presented in Chapter 11.

5 If the PGE is not registered, register the PGE using the **PDPS Science Update Metadata Update** from the **SSIT Manager** GUI.
- The PGE must be registered before a production request can be entered.

6 If it is decided not to re-register the PGE, add the granule that is to be regenerated to the "PDPS Residual Granules" list.

7 From the Planning Workstation, launch the **Production Request Editor** GUI as described in Section 13.2.2.

8 Enter a Production Request (as described in Section 13.2.3) for the relevant PGE/version/profile ID.
- Use **Reprocessing** for the **Processing Type**.
- Use the **DPR Start** and **Stop Time** listed in the Production History for the **Begin** and **End** times.
- View the default PGE runtime parameters and compare them against the runtime parameters obtained from the Production History tar file.
  - Modify the runtime parameter values to match exactly what was used in the original run.

- If granules that need to be regenerated are produced by PGEs that are chained, then the production requests must be entered in that order.
  - For instance, if granules A and B are to regenerated, and PGEs P1 & P2 produce them and if P1 & P2 are chained (P2 takes P1's outputs as its inputs) then the production request for P1 must be entered before entering one for P2.

9   Launch the **Planning Workbench** (refer to Section 13.3.1).
- The **Planning Workbench** is displayed.

10   Create and activate a production plan that includes the newly created production request(s) as described in Section 13.3.2.

11   Return the output ("PDPS Residual Granules") list to the "Data Recovery Procedure for Known Files" procedure in Chapter 17 for further processing.
- Those granules which were not re-generated by this process are listed in the output ("PDPS Residual Granules") list.

Table 13.2-10. Re-Generate Granules Affected by Loss of Files from the Archive - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Identify granules to be reproduced	**read text**
2	Add not-to-be-reproduced granules to the "PDPS Residual Granules "List	**enter text**
3	Retrieve Production History tar file for each element to be reproduced using the **QA Monitor GUI**	Use QA Monitor procedures (Chapter 15)
4	Obtain URs from the QA Monitor GUI	Use QA Monitor procedures (Chapter 15)
5	Extract PGE parameters from the Process Control File	Use QA Monitor procedures (Chapter 15)
6	Launch the **PDPS Operational Metadata GUI** from the **SSIT Manager GUI** if the applicable PGE is not registered	Use SSIT procedure for launching the PDPS Operational Metadata GUI (Chapter 11)
7	Register the PGE	Use SSIT procedure for registering PGEs (Chapter 11)
8	Launch the **Production Request Editor GUI**	Use procedure in Section 13.2.2
9	Enter the **Production Request** for **Reprocessing**	Use procedure in Section 13.2.3
10	Launch the **Planning Workbench**	Use procedure in Section 13.3.1
11	Create and activate a plan for the newly created **Production Request**	Use procedure in Section 13.3.2
12	Return the output ("PDPS Residual Granules") list to the "Data Recovery Procedure for Known Files" procedure (Chapter 17)	

13.3 Creating/Modifying/Deleting Production Strategies

A Production Strategy is a high-level set of priorities by which the Production Planner defines the rules for priorities and preferences in the processing of DPRs. The Production Planner uses the **Production Strategies** GUI to prepare Production Strategies.

Production Strategies work on two levels. First, the Production Planner can update lists of DPR attributes so that each value an attribute can have is tied to a particular priority. For example, the DPR attribute "PR Type" has three values that may have their default priority of (e.g., 2) changed as follows (on a scale of 1 to 10):

- Routine   6
- On-Demand   10
- Reprocessing   4

Next, the Production Planner can change the weight that each attribute's priority is given. For example, weights (from 1 to 100) might be assigned to the DPR attributes as follows:

- PR Type   45
- User Type   15
- PGE Type   20

A weight is also given to the priority specified when the Production Planner created the Production Request as shown in the following example:

- Production Request Editor   20

The total weights assigned to PR Type, User Type, PGE Type and Production Request Editor [Priority] must equal 100.

The values included in the selected strategy are read by the Planning Workbench when prioritizing the DPRs in a production plan.

Finally, there is a Late Start Delta that can be used to increase the priority of all jobs that have been waiting in the Production Queue for more than a day.

Table 13.3-1 provides an Activity Checklist for activities related to the creation of Production Strategies.

Table 13.3-1. Production Strategies - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launch the Production Strategies GUI	(P) 13.3.1	
2	Production Planner	Define or Modify a Production Strategy	(P) 13.3.2	
3	Production Planner	Review the Current Active Strategy	(P) 13.3.3	
4	Production Planner	Delete a Production Strategy	(P) 13.3.4	

13.3.1 Launch the Production Strategies GUI

The Production Strategies GUI is invoked as described in the procedure that follows. Table 13.3-2 presents (in a condensed format) the steps required to launch the Production Strategies GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   In the terminal window, at the command line, enter:

   cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1   - Science Software Integration and Test (SSI&T)
  - TS2   - New Version Checkout
  - OPS   - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3   Set the application environment variables by entering:

   setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4   Start the Production Strategies GUI by entering:

   EcPlProdStratStart <MODE>

- The **Production Strategies** GUI is launched.

Table 13.3-2. Launch the Production Strategies GUI - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables if necessary	**enter text, press Enter**
4	**EcPlProdStratStart <MODE>**	**enter text, press Enter**

13.3.2 Define or Modify a Production Strategy

The Production Planner uses the Production Strategies GUI to develop Production Strategies. Table 13.3-3 presents (in a condensed format) the steps required to define or modify a production strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Strategies** GUI (refer to Section 13.3.1).
- The **Production Strategies** GUI is displayed.

2   If defining a new production strategy, execute the following menu path:

File → New
- The fields of the **Production Strategies** GUI are reset.

3   If modifying an existing production strategy, first **single-click** the option button associated with the **Production Strategies** field, then highlight (in the option menu) the name of the production strategy to be modified.
- Data pertaining to the selected production strategy are displayed in the applicable fields of the **Production Strategies** GUI.
- Alternatively, it is possible to execute the following menu path: **File → Open**, select the desired production strategy from the list on the **Open** window, and **single-click** on the **Ok** button to open the production strategy.

NOTE: The **Help** button at the bottom of **Production Strategies Open** window is non-functional (does not work).

4   If changing the default priority for PR Type, in the **Default** field below the **PR Type** button enter:

<default value>
- The range for the default is from 1 to 10.

5   If changing the default priority for User Type, in the **Default** field below the **User Type** button enter:

<default value>

6   If changing the default priority for PGE Type, in the **Default** field below the **PGE Type** button enter:

<default value>

7   If defining or modifying a priority for a type of production request, first **single-click** on the **PR Type** button.
- The different types of production requests are displayed in the **Type List** field at the bottom left of the GUI.
- Options are: **Routine**, **OnDemand**, and **Reprocessing**.

8   If defining a priority for a type of production request **not** currently listed in the **PR Type Value-Priority** list, **single-click** on that PR type in the **Type List** field.
- The PR type is highlighted.
- It is possible to highlight multiple PR types (by **single-clicking** on each one in turn) if they are all going to be assigned the same priority.

9   If redefining or deleting a priority for a type of production request **already** listed in the **PR Type Value-Priority** list, **single-click** on that PR type in the **Value-Priority** list.
- The PR type is highlighted.
- It is possible to highlight multiple PR types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key) if the same action is going to be taken with respect to all of them.
- The **Find** button provides a means of performing a keyword search of the **PR Type Value-Priority** list.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PR Type Value-Priority** list that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

10   If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.
- Alternatively, in the **Priority** field enter:
  <priority value>
- The acceptable range for the priority is from 1 to 10.

11   **Single-click** on the appropriate button from the following selections:
- **Add** - to approve a priority for an additional PR type and display the selected PR type and priority in the **PR Type Value-Priority** list at the left center of the GUI.
- **Modify** - to approve a revised priority for the selected PR type and display the PR type and modified priority in the **PR Type Value-Priority** list.
- **Delete** - to delete the priority for the selected PR type and remove the PR type and priority from the **PR Type Value-Priority** list.

12   Repeat Steps 7 through 11 as necessary until all PR Type priorities (as shown in the **PR Type Value-Priority** field) are correct.

13   If defining or modifying a priority for a type of user, first **single-click** on the **User Type** button.
  - The different types of users are displayed in the **Type List** field at the bottom left of the GUI.
  - Options are: **Operator**, **DAAC Manager**, **Scientist**, and **Researcher**.

14   If defining a priority for a type of user **not** currently listed in the **User Type Value-Priority** list, **single-click** on that user type in the **Type List** field.
  - The user type is highlighted.
  - It is possible to highlight multiple user types (by **single-clicking** on each one in turn).

15   If redefining or deleting a priority for a user type **already** listed in the **User Type Value-Priority** list, **single-click** on that user type in the **Value-Priority** list.
  - The user type is highlighted.
  - It is possible to highlight multiple user types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key).
  - The **Find** button provides a means of performing a keyword search of the **User Type Value-Priority** list.
    - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **User Type Value-Priority** list that has the search text is highlighted.)
    - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

16   If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.
  - Alternatively, in the **Priority** field enter:

    <priority value>
  - The acceptable range for the priority is from 1 to 10.

17   **Single-click** on the appropriate button from the following selections:
  - **Add** - to approve a priority for an additional user type and display the selected user type and priority in the **User Type Value-Priority** list near the center of the GUI.
  - **Modify** - to approve a revised priority for the selected user type and display the user type and modified priority in the **User Type Value-Priority** list.
  - **Delete** - to delete the priority for the selected user type and remove the user type and priority from the **User Type Value-Priority** list.

18   Repeat Steps 13 through 17 as necessary until all user type priorities (as shown in the **User Type Value-Priority** field) are correct.

19   If defining a priority for a type of PGE, first **single-click** on the **PGE Type** button.
- The different types of PGEs are displayed in the **Type List** field at the bottom left of the GUI.
- All PGEs currently registered in the PDPS database (for the mode in which the **Production Strategies** GUI was brought up) are listed.

20   If defining a priority for a type of PGE **not** currently listed in the **PGE Type Value-Priority** list, **single-click** on that PGE type in the **Type List** field.
- The PGE type is highlighted.
- It is possible to highlight multiple PGE types (by **single-clicking** on each one in turn).

21   If redefining or deleting a priority for a PGE type **already** listed in the **PGE Type Value-Priority** list, **single-click** on that PGE type in the **Value-Priority** list.
- The PGE type is highlighted.
- It is possible to highlight multiple PGE types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key).
- The **Find** button provides a means of performing a keyword search of the **PGE Type Value-Priority** list.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PGE Type Value-Priority** list that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

22   If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.
- Alternatively, in the **Priority** field enter:

  <priority value>

- The acceptable range for the priority is from 1 to 10.

23   **Single-click** on the appropriate button from the following selections:
- **Add** - to approve a priority for an additional PGE type and display the selected PGE type and priority in the **PGE Type Value-Priority** list near the center of the GUI.
- **Modify** - to approve a revised priority for the selected PGE type and display the PGE type and modified priority in the **PGE Type Value-Priority** list.
- **Delete** - to delete the priority for the selected PGE type and remove the PGE type and priority from the **PGE Type Value-Priority** list.

24   Repeat Steps 19 through 23 as necessary until all PGE type priorities (as shown in the **PGE Type Value-Priority** field) are correct.

25   In the **Weight** field below the **PR Type** button enter:

<weight>

- The acceptable range for weights is from 1 to 100.
- The **Total Weight** field displays updated totals of all weighting factors as they are entered.
- When entering weights for the **PR Type**, **User Type**, **PGE Type**, and **Production Request Editor** factors, relative values can be entered without regard to whether the values in the four categories add up to 100. The **Normalize** button provides a means of eventually ensuring that the total of all four categories equals 100.
- The assigned weight in each category is multiplied by the priority for each type. To maintain a high priority (low number, such as one), assign a low weight; to ensure a low priority, assign a relatively high weight.

26   In the **Weight** field below the **User Type** button enter:

<weight>

27   In the **Weight** field below the **PGE Type** button enter:

<weight>

28   In the **Production Request Editor** field enter:

<weight>

- The priority to which the weight is applied is the priority assigned using the Production Request Editor when a production request is created.

29   **Single-click** on the **Normalize** button.
- The Planning Subsystem adjusts all weighting factors to produce a total weight of 100 (as displayed in the **Total Weight** field).

30   If it is necessary to change the priority of jobs that have been waiting in the Production Queue for more than a day, in the **Late Start Delta Priority** field enter:

<priority value>

- The range for the Late Start Delta Priority is from 1 to 100.

31   To start the process of saving the production strategy execute the following menu path:

File → Save As

- A **Save As** window similar to the **File Selection** window is displayed.

32   In the **Save As** field enter:

<strategy file name>

33   **Single-click** on the appropriate button from the following selections:

- **OK** - to accept the file name in the **Save As** field.
  - The **Save As** window is dismissed.
  - The production strategy is saved with the specified file name.
- **Cancel** - to dismiss the. **Save As** window without saving the production strategy.

NOTE: The **Help** button at the bottom of **Production Strategies Save As** window is non-functional (does not work).

34   To start the process of modifying or creating another production strategy execute the following menu path:

File → New

- Return to Step 2 to create another new production strategy.
- Return to Step 3 to modify an existing production strategy.

35   To start the process of exiting from the **Production Strategies** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

36   **Single-click** on the appropriate button from the following selections:

- **OK** - to exit from the **Production Strategies** GUI.
  - The **Production Strategies** GUI is dismissed.
- **Cancel** - to return to the **Production Strategies** GUI.

Table 13.3-3. Define or Modify a Production Strategy - Quick-Step Procedures
(1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**File → New**	**single-click**
3	**<default priority>** (for **PR Type**) (if applicable)	**enter text**
4	**<default priority>** (for **User Type**) (if applicable)	**enter text**
5	**<default priority>** (for **PGE Type**) (if applicable)	**enter text**
6	**PR Type** button	**single-click**
7	**<PR type>** (from the **Type List** field)	**single-click**
8	**<priority value>** (**Priority** field)	**enter text** or **single-click** as necessary
9	**Add** button	**single-click**
10	Repeat Steps 7 through 9 for additional PR types if applicable	
11	**User Type** button	**single-click**
12	**<user type>** (**Type List** field)	**single-click**

Step	What to Enter or Select	Action to Take
13	**<priority value>** (**Priority** field)	**enter text** or **single-click** as necessary
14	**Add** button	**single-click**
15	Repeat Steps 12 through 14 for additional user types if applicable	
16	Select **PGE Type** button	**single-click**
17	Select **<PGE type>** from the **Type List** field	**single-click**
18	Enter **<priority value>** in the **Priority** field	**enter text** or **single-click** as necessary
19	Select **Add**	**single-click**
20	Repeat Steps 17 through 19 for additional PGE types if applicable	
21	**<weight>** (**PR Type**)	**enter text**
22	**<weight>** (**User Type**)	**enter text**
23	**<weight>** (**PGE Type**)	**enter text**
24	**<weight>** (**Production Request Editor**)	**enter text**
25	**Normalize** button	**single-click**
26	**<priority value>** (for **Late Start Delta Priority**) (if applicable)	**enter text**
27	**File → Save As**	**single-click**
28	**<strategy file name>** (**Save As** field)	**enter text**
29	**Ok** button	**single-click**
30	**File → Exit** (when applicable)	**single-click**
31	**OK** button (when applicable)	**single-click**

13.3.3 Review the Current Active Strategy

The Production Planner uses the Production Strategies GUI to review the production strategy applied to the currently active production plan. Table 13.3-4 presents (in a condensed format) the steps required to review the current active strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Strategies** GUI (refer to Section 13.3.1).
- The **Production Strategies** GUI is displayed.

2   Execute the following menu path:

   Options → activeStrategy

- The **Active Production Strategy** window is displayed.

3   Review the data displayed in the **Active Production Strategy** window:

- **PR Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual PR types and values if any are listed.]
- **User Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual user types and values if any are listed.]
- **PGE Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual PGE types and values if any are listed.]
- **Production Request Editor Weight.**
- **Late Start Delta Priority.**

4   To exit from the **Active Production Strategy** window and return to the **Production Strategies** GUI **single-click** on the **Cancel** button.

- The **Active Production Strategy** window is dismissed.
- The **Production Strategies** GUI is displayed.

Table 13.3-4. Review the Current Active Strategy - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**Options → activeStrategy**	**single-click**
3	Observe the production strategy information	**read text**
4	**Cancel** button	**single-click**

13.3.4 Delete a Production Strategy

The Production Planner uses the Production Strategies GUI to delete production strategies that are no longer needed. Table 13.3-5 presents (in a condensed format) the steps required to delete a production strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Production Strategies** GUI (refer to Section 13.3.1).

- The **Production Strategies** GUI is displayed.

2   **Single-click** the option button associated with the **Production Strategies** field, then highlight (in the option menu) the name of the production strategy to be deleted.

- Data pertaining to the selected production strategy are displayed in the applicable fields of the **Production Strategies** GUI.
- Alternatively, it is possible to execute the following menu path: **File → Open**, select the desired production strategy from the list on the **Open** window, and **single-click** on the **Ok** button to open the production strategy.

3   Execute the following menu path:

Edit → Delete

- A production strategy deletion confirmation dialogue box is displayed requesting confirmation of the decision to delete the production strategy.

4   Click on the appropriate button from the following selections:

- **OK** - to confirm deletion of the production strategy and dismiss the dialogue box.
    - The deletion confirmation dialogue box is dismissed.
    - The production strategy is deleted.
- **Cancel** - to preserve the production strategy and dismiss the dialogue box.
    - The deletion confirmation dialogue box is dismissed.
    - The production strategy is not deleted.

5   To start the process of exiting from the **Production Strategies** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

6   **Single-click** on the appropriate button from the following selections:

- **OK** - to exit from the **Production Strategies** GUI.
    - The **Production Strategies** GUI is dismissed.
- **Cancel** - to return to the **Production Strategies** GUI.

Table 13.3-5. Delete a Production Strategy - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**<stra0tegy name>** (from **Production Strategies** option list)	**single-click**
3	**Edit → Delete**	**single-click**
4	**OK** button	**single-click**
5	**File → Exit** (when applicable)	**single-click**
6	**OK** button	**single-click**

13.4 Creating/Deleting Production Plans

The planning process involves the Production Planner preparing monthly and weekly production plans as well as a daily production schedule from the most current weekly plan. Although production planning varies from DAAC to DAAC, the following guidelines are generally applicable:

- Monthly plans.
  - Developed for the coming month and one or two months in advance.
  - Produced, reviewed, updated, published and distributed approximately two weeks before the beginning of the month.
  - Plan for the coming month is used to establish a baseline against which production targets can be measured.
- Weekly plans.
  - Produced, reviewed, updated, published and distributed approximately five days before the beginning of the coming week.
  - Used to produce a baseline for comparison of planned vs. actual production results.
- Daily plan or schedule.
  - Produced each day for the next processing day.
  - Developed from the current weekly plan, adjusted to reflect the actual processing accomplished and the actual resources available at the time the daily schedule is generated.
  - The daily plan is the type of plan that is implemented through the **Planning Workbench** GUI.
  - If there is a very large processing volume, it may be advisable to divide the daily plan into multiple plans; e.g., one for each shift or one for each four-hour period.

During normal processing, when reasonably accurate predictions of the processing time for the PGEs are available, the processing schedule should result in a reasonably accurate prediction of when data products will be generated. However, during abnormal situations (e.g., equipment failure), what is actually accomplished could depart significantly from the plan. In such situations, the Production Planner may choose to develop new plans to reflect current events. This process is known as "replanning."

Table 13.4-1 provides an Activity Checklist for activities related to the creation of Production Plans.

Table 13.4-1. Production Plans - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launch the Planning Workbench and Planning Timeline GUIs	(P) 13.4.1	
2	Production Planner	Create a New Production Plan	(P) 13.4.2	
3	Production Planner	Delete a Production Plan	(P) 13.4.3	
4	Production Planner	Review a Plan Timeline	(P) 13.4.4	

13.4.1 Launch the Planning Workbench and Planning Timeline GUIs

The Planning Workbench and Planning Timeline GUIs are invoked from a UNIX command line prompt. Table 13.4-2 presents (in a condensed format) the steps required to launch the Planning Workbench and Planning Timeline GUIs. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Planning/Management Workstation host.
  - Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
  - For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

  - **<MODE>** is current mode of operation.
    - − TS1   - Science Software Integration and Test (SSI&T)
    - − TS2   - New Version Checkout
    - − OPS   - Normal Operations
  - "utilities" is the directory containing the Planning Subsystem start-up scripts.

3   Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

  - Application home environment is entered
  - When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

NOTE: The selection of a start-up script for launching the **Planning Workbench** GUI depends on the number of DPRs in the PDPS database. If the number is very high (~4000), the **EcPlSomeStart** script is used to start the underlying processes (Message Handler, System Name Server, and Resource Model). Then additional scripts are used to start the **Planning Workbench** GUI and the **Production**

Planning Timeline GUI. If the number of DPRs in the PDPS database is not very high (less than about 4000), the **EcPlAllStart** script is used to start the underlying processes, **Planning Workbench** GUI and **Production Planning Timeline** GUI.

4   If the number of DPRs in the PDPS database is not large (less than about 4000), start the Planning Workbench-related applications by entering:

EcPlAllStart <MODE> <Application_id>

- The System Name Server and Resource Model are launched.
- The **Message Handler** GUI is displayed.
- Eventually, the **Planning Workbench** GUI is displayed.
- Then the **Production Planning Timeline** GUI is displayed.
    - The **Production Planning Timeline** GUI usually occupies the entire screen when it is initially displayed.
- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a production planning session.
- Go to Step 8.

5   If the number of DPRs in the database is large (~4000 or more), start the Message Handler, System Name Server, and Resource Model by entering:

EcPlSomeStart <MODE> <Application_id>

- The **Message Handler** GUI is displayed.
- Wait until the Resource Model is completely up before proceeding with starting the **Planning Workbench** GUI.

6   If the number of DPRs in the database is large (~4000 or more), start the **Planning Workbench** GUI by entering:

EcPlWbStart <MODE> <Application_id>

- The **Planning Workbench** GUI is displayed.

7   If the number of DPRs in the database is large (~4000 or more), start the **Production Planning Timeline** GUI by entering

EcPlTlStart <MODE> <Application_id>

- The **Production Planning Timeline** GUI is displayed.

8   If the **Production Planning Timeline** GUI is occupying the whole screen, either **single-click** on the "minimize" icon in the upper right corner of the GUI or adjust the window size and the view of the timeline as necessary using the mouse.
- Grab a corner of the timeline window with the cursor and resize the window as desired.

Table 13.4-2. Launch the Planning Workbench and Planning Timeline GUIs -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set environment variables if necessary	**enter text, press Enter**
4	**EcPlAllStart <MODE> <application_id>**	**enter text, press Enter**

13.4.2 Create a New Production Plan

The Production Planner uses the Planning Workbench when creating a plan for production data processing at the DAAC. The Planning Workbench GUI provides the means by which the Production Planner selects specific PRs whose DPRs are to be run. The planning tool provides a forecast of the start and completion times of the jobs based upon historical experience in running these PGEs (as determined during the SSI&T process). Through the planning tool, when the generated plan is "activated," the information included in the plan is transferred to the Data Processing Subsystem and loaded into the Platinum AutoSys tool where production processing is managed.

Table 13.4-3 presents (in a condensed format) the steps required to create a new Production Plan. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Launch the Planning Workbench (refer to Section 13.4.1).
- The **Planning Workbench** and **Planning Timeline** GUIs are displayed.

2   Execute the following menu path on the **Planning Workbench GUI**:

File → New

- The **New Plan** window is displayed.

3   In the **Plan Names** field enter:

<plan name>

- Name is displayed in **Plan Names** field.
- Status is **Candidate**.

4   **Single-click** on the appropriate button from the following selections:
- **Ok** - to accept the file name in the **Plan Names** field.
  - The **New Plan** window is dismissed.
  - The production plan is saved with the specified file name.
- **Apply** - to save the production plan without dismissing the **New Plan** window.
  - The production plan is saved with the specified file name.

- **Cancel** - to dismiss the **New Plan** window without saving the production plan.

NOTE: The **Help** buttons at the bottom of various **Planning Workbench** windows are non-functional (do not work).

5 If applicable, **single-click** on the option button associated with the **Strategy** field, then select the desired production strategy from the option menu.

6 If applicable, in the **Comment** field enter:

   <comment>

7 Select a Production Request to be scheduled by **single-clicking** on the request line and **single-clicking** on **schedule/unschedule** up and down arrows to move the Production Request from the **Unscheduled** panel to the **Scheduled** panel or vice versa.
- Production Request will appear in the appropriate panel.
- Operator can **single-click** on multiple Production Requests to add them all at the same time.
- If the PR depends upon another PR that is not scheduled, then the following message is displayed: "Production Request '*xxx*' must be selected before DPR '*yyy*' of Production Request '*zzz*' can be scheduled."
- If all DPRs associated with a PR have been run, the PR cannot be rescheduled.
- If processing of the currently active plan is to be continued when the new plan is activated, include the PR(s) for the currently active plan in the new plan.
- In the **Scheduled** list, items with the prefix "GE_" are resource reservations (also called "ground events").
- All ground events are automatically scheduled with any plan; therefore, the ground events are normally displayed in the Scheduled list.
- Whenever a plan is activated, the ground events are activated as well as the DPRs associated with the specified PRs.
- If a ground event appears in the **Unscheduled** list, the ground event has lost allocations.

8 If the priority of any PR in the **Scheduled** list needs to be changed, perform Steps 9 through 13; otherwise go to Step 14.

9 **Single-click** on the PR entry in the **Scheduled** list to highlight it.

10 **Single-click** on the **Prioritize** button.
- The **Priority popup** window is displayed.

11 In the **Production Request(s) priority** field enter:

   <priority value>

12   **Single-click** on the appropriate button from the following selections:
- **OK** - to accept the new priority in the **Production Request(s) priority** field.
  – The **Priority popup** window is dismissed.
  – The new priority for the PR is saved in the database.
- **Cancel** - to dismiss the **Priority popup** window without saving the new priority.

13   Repeat Steps 9 through 12 for any additional PR(s) needing a change of priority.

14   Execute the following menu path:

File → Save As

- The **Save Plan** window is displayed.

15   If the **plan name** for the production plan is not displayed in the **Plan Names** field, enter:

<plan name>

16   **Single-click** on the appropriate button from the following selections:
- **Ok** - to accept the file name in the **Plan Names** field.
  – The **Save Plan** window is dismissed.
  – The production plan is saved with the specified file name.
  – The **Planning Workbench** GUI is displayed.
  – The **Plan Name** is displayed.
  – The **Status** displayed is **Candidate**.
- **Apply** - to save the production plan without dismissing the **Save Plan** window.
  – The production plan is saved with the specified file name.
- **Cancel** - to dismiss the **Save Plan** window without saving the production plan.

17   If the plan is to be activated immediately, perform Steps 18 and 19; otherwise go to Step 20.

18   **Single-click** on the **Activate** button.
- A **Confirm Activation** dialogue box is displayed.

19   **Single-click** on the appropriate button from the following selections:
- **Yes** - to activate the plan.
  – The **Confirm Activation** dialogue box is dismissed.
  – The new plan is activated.
  – The time of plan activation is displayed next to **Rollover Time** on the **Planning Workbench** GUI.
  – When a plan is activated the Data Processing Requests (DPRs) associated with the planned PRs are transferred to Job Management in the Data Processing Subsystem.
  – Once its data dependencies have been satisfied, each DPR is "released" to be run as processing resources become available.

- Activating a new plan causes the current active plan to get "replanned over" by the selected plan.
- **No** – to dismiss the **Confirm Activation** dialogue box without activating the plan.

20   If it is desired to baseline the plan, perform Steps 21 through 25; otherwise go to Step 26.

21   **Single-click** on the **Baseline** button.
- A confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** is displayed.
- Clicking on the **Baseline** button records the plan and the time of baselining.
- A baseline plan can be used as a point of comparison with which to compare future plans and results.

22   If it is desired to baseline the plan, **single-click** on the appropriate button from the following selections:
- **Yes** - to baseline the plan.
  - The confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** is dismissed.
  - The plan is baselined.
- **No** – to dismiss the confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** without baselining the plan.

23   Execute the following menu path:

File → Save As

- The **Save Plan** window is displayed.

24   If the **plan name** for the production plan is not displayed in the **Plan Names** field, enter:

<plan name>

25   **Single-click** on the appropriate button from the following selections:
- **Ok** - to accept the file name in the **Plan Names** field.
  - The **Save Plan** window is dismissed.
  - The production plan is saved with the specified file name.
- **Apply** - to save the production plan without dismissing the **Save Plan** window.
  - The production plan is saved with the specified file name.
- **Cancel** - to dismiss the **Save Plan** window without saving the production plan as a baseline plan.

26   Repeat Steps 1 through 25 to perform additional production planning activities.

27   To view the **Planning Timeline** perform the procedure in Section 13.4.3.

28   To quit the **Planning Workbench** GUI when production planning is complete execute the following menu path:

File → Exit

29   After quitting the **Planning Workbench** GUI **single-click** in the UNIX window used to start the **Planning Workbench** GUI.
- The Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the Planning Workbench if necessary.

30   Shut down Planning Workbench-related applications by entering:

EcPlSlayAll <MODE> <Application_id>

- The following Planning Workbench-related applications shut down:
    – Planning Workbench (if it has not already been shut down).
    – Planning Timeline (if it has not already been shut down).
    – Message Handler.
    – System Name Server.
    – Resource Model.

31   Obtain a list of active processes in the specified mode by entering:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

ps -auxwww | grep <MODE>

32   Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.
- None of the following processes should be active:
    – EcPlWb
    – EcPlTl
    – EcPlMsh
    – EcPlSns
    – EcPlRm

33   If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>

34   Repeat Steps 31 through 33 as necessary.

Table 13.4-3. Create a New Production Plan - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench**	Use procedure in Section 13.4.1
2	**File → New**	**single-click**
3	**<plan name>**	**enter text**
4	**Ok** button	**single-click**
5	**<strategy>** (from **Strategy** button)	**single-click**
6	**<Production Request(s)> to schedule/unschedule**	**single-click**
7	**Either Schedule button or Unschedule button as applicable**	**single-click**
8	**File Save As**	**single-click**
9	**<plan name>**	**enter text**
10	**Ok button**	**single-click**
11	**Activate button (if applicable)**	**single-click**
12	**Yes button (if applicable)**	**single-click**
13	**Baseline button (if applicable)**	**single-click**
14	**Yes button (if applicable)**	**single-click**
15	**File Save As (if applicable)**	**single-click**
16	**<plan name> (if applicable)**	**enter text**
17	**Ok button (if applicable)**	**single-click**
18	**View the Planning Timeline if desired**	**Use procedure in Section 13.4.3**
19	**File Exit (when applicable)**	**single-click**
20	**EcPlSlayAll <MODE> <Application_id>**	**enter text, press Enter**
21	**ps -ef	grep <MODE>**	**enter text, press Enter**
22	**kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx> to terminate active process(es) (if necessary)**	**enter text, press Enter**

13.4.3 Delete a Production Plan

The Production Planner uses the Planning Workbench GUI to delete production plans that are no longer needed.   Table 13.4-4 presents (in a condensed format) the steps required to delete a production plan. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Planning Workbench** GUI (refer to Section 13.4.1).
- The **Planning Workbench** GUI is displayed.

2   Execute the following menu path:

File → Open

- The **Open Plan** window is displayed.
- A list of Production Plans is displayed in the **Open Plan** window.

3   Select (highlight) the production plan to be deleted by **single-clicking** on the corresponding name in the list of plans.

- The plan name is displayed in the **Plan Names** field of the **Open Plan** window.

4   **Single-click** on the **OK** button.

- The **Open Plan** window is dismissed.
- The plan information is displayed on the **Planning Workbench** GUI.

5   Execute the following menu path:

File → Delete

- The production plan is deleted.

6   To quit the **Planning Workbench** GUI execute the following menu path:

File → Exit

7   After quitting the **Planning Workbench** GUI **single-click** in the UNIX window used to start the **Planning Workbench** GUI.

- The Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the Planning Workbench if necessary.

8   Shut down Planning Workbench-related applications by entering:

EcPlSlayAll <MODE> <Application_id>

- The following Planning Workbench-related applications shut down:
    - Planning Workbench (if it has not already been shut down).
    - Planning Timeline (if it has not already been shut down).
    - Message Handler.
    - System Name Server.
    - Resource Model.

9   Obtain a list of active processes in the specified mode by entering:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

    ps -auxwww | grep <MODE>

10   Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.
- None of the following processes should be active:
  - EcPlWb
  - EcPlTl
  - EcPlMsh
  - EcPlSns
  - EcPlRm

11   If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>

12   Repeat Steps 9 through 11 as necessary.

Table 13.4-4. Delete a Production Plan - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench** GUI	Use procedure in Section 13.4.1
2	**File → Open**	**single-click**
3	**<plan name>** (from **Production Requests** list)	**single-click**
4	**OK** button	**single-click**
5	**File→Delete**	**single-click**
6	**File → Exit**	**single-click**
7	**EcPlSlayAll <MODE> <Application_id>**	**enter text, press Enter**
8	**ps -ef	grep <MODE>**	**enter text, press Enter**
9	**kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>** to terminate active process(es) (if necessary)	**enter text, press Enter**

13.4.4 Review a Plan Timeline

Table 13.4-5resents (in a condensed format) the steps required to review planning timelines. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Launch the **Planning Workbench** and **Timeline** GUIs (refer to Section 13.4.1).
- The **Planning Timeline** GUI is displayed.

- If you have previously saved a configuration file, you may load it by executing the following menu path:

  File → Load Configuration

  − Change directory in GUI box, if necessary.
- Otherwise continue with Step 2.

2   Execute the following menu path:

File → Open Plan

- The **Open Plan** window is displayed with a list of plans.

3   **Single-click** on the name (in the **Items** list) of the plan to be reviewed.

4   **Single-click** on the appropriate button from the following selections:
- **OK** - to open the selected plan and dismiss the **Open Plan** window.
  − The timeline for the specific plan is displayed.
  − Name is displayed in the Title bar.
- **Apply** - to open the selected plan without dismissing the **Open Plan** window.
  − The timeline for the specific plan is displayed.
  − Name is displayed in the Title bar.
- **Cancel** - to dismiss the **Open Plan** window without opening any plan.

NOTE: The **Help** buttons at the bottom of various **Production Planning Master Timeline** windows are non-functional (do not work).

5   Observe the production plan information displayed on the timeline GUI.

6   If a different time scale (start and end dates and times) is desired, perform Steps 7 through 10; otherwise, go to Step 11.

7   Execute the following menu path:

Time → Change Plan Window

- The **plan window edit** window appears with default times.

8   In the **Plan Win Start** fields enter the start date and time in the following formats:

<DD MMM YYYY> <hh:mm:ss>

9   In the **Plan Win End** fields enter the end date and time in the following formats:

<DD MMM YYYY> <hh:mm:ss>

10   **Single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **plan window edit** window.

- **Apply** - to accept the changes without dismissing the **plan window edit** window.
- **Cancel** - to cancel the changes and dismiss the **plan window edit** window.

11   If a different time span is desired, **single-click** on the **Show** pushbutton and select one of 11 time increments between **5 min** and **168 hrs** for the timeline scale.
- The entry "**other**" has no purpose at this time.

12   If no resources are displayed on the GUI or if different resources should be displayed, perform Steps 13 through 19; otherwise, go to Step 20.

13   Execute the following menu path:

Display → Change resources

- **Resource edit** window with lists of **Available Resources** and **Viewed Resources** is displayed.

14   **Single-click** resource(s) in desired list.
- Multiple resources may be selected by single-clicking each desired resource.

15   **Single-click** either **Add** or **Del**.
- **Add** to move the resource(s) from the **Available** list to the **Viewed** list.
- **Del** to remove items from the **Viewed** List.

16   To change the order in which resources are displayed on the timeline, **single-click** on an item in the **Viewed Resources** list.

17   To change the order in which resources are displayed on the timeline, **single-click** on the **up** or **down** arrow(s) as appropriate.
- Selected resource moves up or down in order on the list.

18   Repeat Steps 16 and 17 as necessary.

19   **Single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **Resource edit** window.
- **Apply** - to accept the changes without dismissing the **Resource edit** window.
- **Cancel** - to cancel the changes and dismiss the **Resource edit** window.

20   If different color-coding of the timeline is desired, perform Steps 21 through 25; otherwise, go to Step 26.

21   Execute the following menu path:

Display → Change colors

- Color grid appears with a list of Production Requests.

22 **Single-click** on the Production Request name.

23 **Single-click** on the desired color for the Production Request.
- New color appears on the horizontal bar between color and Production Request selections.

24 Repeat Steps 22 and 23 as necessary.

25 **Single-click** on the appropriate button from the following selections:
- **OK** - to accept the changes and dismiss the **Color Selections** window.
- **Apply** - to accept the changes without dismissing the **Color Selections** window.
- **Cancel** - to cancel the changes and dismiss the **Color Selections** window.

26 Observe the production plan information displayed on the timeline GUI.

27 If desired, save the current configuration as a file by executing the following menu path:

  File → Save Configuration

28 **Single-click** on the appropriate button from the following selections:
- **OK** - to save the configuration and dismiss the **Color Selections** window.
- **Apply** - to save the configuration without dismissing the **Color Selections** window.
- **Cancel** - to dismiss the **Color Selections** window without saving the configuration.

29 To exit the **Planning Master Timeline** GUI, execute the following menu path:

File → Exit
- The **Planning Master Timeline** GUI is dismissed.

Table 13.4-5. Review a Plan Timeline - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench** and **Timeline**	Use procedure in Section 13.4.1
2	**File → Load Configuration**) (optional)	**single-click**
3	**File → Open Plan**	**single-click**
4	**<plan name>**	**single-click**
5	**Ok** button	**single-click**
6	Observe the production plan information	**read text**
7	**Time → Change Plan Window**	**single-click**
8	**<DD MMM YYYY> <hh:mm:ss>** (in **Plan Win Start** fields)	**enter text**
9	**<DD MMM YYYY> <hh:mm:ss>** (in **Plan Win End** fields)	**enter text**
10	**Ok** button	**single-click**
11	**<time span>** (from **Show** pushbutton)	**single-click**

Table 13.4-5. Review a Plan Timeline - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
12	**Display → Change resources** (if applicable)	**single-click**
13	**<resources>** (if applicable)	**single-click**
14	**Add** button (if applicable)	**single-click**
15	**<viewed resource>** to be reordered (if applicable)	**single-click**
16	**<up>** arrow or **<down>** arrow (as necessary to reorder viewed resources) (if applicable)	**single-click**
17	**Ok** button (if applicable)	**single-click**
18	**Display → Change colors** (if applicable)	**single-click**
19	**<Production Request>** (if applicable)	**single-click**
20	**<color>** (for Production Request) (if applicable)	**single-click**
21	**Ok** button (if applicable)	**single-click**
22	Observe the production plan information	**read text**
23	**File → Save Configuration** (if applicable)	**single-click**
24	**<file name>** (if applicable)	**enter text**
25	**Ok** button (if applicable)	**single-click**
26	**File → Exit** (when applicable)	**single-click**

13.5 Cleaning the PDPS Database and Science Processing Disks

At DAACs with a heavy data processing load it is essential to perform PDPS database cleanup and virtual-computer "garbage collection" on a regular basis. The frequency of cleanup is determined by each DAAC's needs. Failure to perform cleanup is likely to lead to performance problems with the Planning and Data Processing Subsystems.

It is recommended that the database cleanup and "garbage collection" processes be run in the following order:

- Database cleanup.
- "Garbage collection" on the virtual computer(s).
- Database cleanup (again).

If not already done, it is possible to create a script to run the preceding processes. If it is necessary to run particular scripts individually, refer to the sections that follow.

NOTE: To the extent possible the scripts mentioned in this section should be run when the system is relatively quiet; i.e., when no PRs/DPRs are being created and no jobs are running in AutoSys.

Table 13.5-1 provides an Activity Checklist for activities related to cleaning the PDPS database and science processing disks.

Table 13.5-1. Cleaning the PDPS Database and Science Processing Disks - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner Database Administrator	Clean the PDPS Database	(P) 13.5.1	
2	Production Planner Production Monitor	Perform Garbage Collection	(P) 13.5.2	
3	Production Planner Production Monitor	Run the Deletion Server Client	(P) 13.5.3	
4	Production Planner Production Monitor	Resolve PDPS Database and Science Processing Disk Content Discrepancies	(P) 13.5.4	
5	Production Planner Database Administrator	Save and/or Reset the PDPS Database	(P) 13.5.5	

13.5.1 Clean the PDPS Database

In the /usr/ecs/<MODE>/CUSTOM/utilities directory on the Planning/Management Workstation there is a script that can be run to clean up some tables in a PDPS database. When it runs, the script tries to delete applicable records in the following order:

- Data Processing Requests based on timeStamp completionState(SUCC_DEL).
- Production Requests that have no associated DPRs.
- Dynamic data granules that are not used by any DPR or by the Data Processing Subsystem.
- List of data granules that are not deleted because of Data Processing Subsystem usage.
- PGEs marked with a deleteFlag.
- Science Software that has no associated PGE.

The database cleaning script is invoked from a UNIX command line prompt. Table 13.5-2 presents (in a condensed format) the steps required to clean the PDPS database. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line prompt, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
    - TS1   - Science Software Integration and Test (SSI&T)
    - TS2   - New Version Checkout
    - OPS   - Normal Operations
- "utilities" is the directory containing the Planning Subsystem utility/start-up scripts.

4 Start the database cleaning script by entering:

EcPlDbClean <MODE> <database user> <database password> <database server> <months> <days>

- **<database user>** is the user name for logging in to interactive structured query language (isql).
- **<database password>** is the password for isql login.
- **<database server>** refers to the name of the PDPS database server (e.g., x0sps02_srvr).
- **<months>** is a number specifying the removal of records that are older than that number of months.
- **<days>** is an optional argument. It is a number that specifies the removal of records that are older than that number of days.
- Both **<months>** and **<days>** are taken into account by the cleaning script.
- When the process has finished, the following type of message is displayed:

    Log is written to /usr/ecs/OPS/CUSTOM/logs/EcPlDbClean.log

5 When the EcPlDbClean process has finished, enter:

cd ../logs

- Change directory to the directory containing the production planning log files (including EcPlDbClean.log).

6 At the command line prompt enter:

pg <filename>

- **<filename>** refers to the log file to be reviewed (e.g., EcPlDbClean.log).
- The first page of the log file is displayed.

- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **view, vi**) can be used to review the log file.

7   Review the log file to determine the results of running the script.
- To exit from **pg** at the **:** prompt enter:

q
   − The command line prompt is displayed.

Table 13.5-2. Clean the PDPS Database - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcPlDbClean <MODE> <database server> <database user> <database password> <months> <days>**	**enter text, press Enter**
5	**cd ../logs** (when the EcPlDbClean process has finished)	**enter text, press Enter**
6	**pg <filename>**	**enter text, press Enter**
7	Review the log file	**read text**

13.5.2 Perform Garbage Collection

In the /usr/ecs/*MODE*/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrGarbageCollectorStart) that can be run to delete unneeded files from the science processing disks and update the PDPS database accordingly.

The actual executable invoked by the EcDpPrGarbageCollectorStart script is EcDpPrDeletionClient. However, the EcDpPrGarbageCollectorStart script differs from the EcDpPrDeletionClientStart script in the following three ways:

- Does not open a separate xterm window.
- Requires specification of a MACHINE_TO_COLLECT variable.
- Includes retry logic in case of database deadlock.

The garbage-collection script is invoked from a UNIX command line prompt. Table 13.5-3 presents (in a condensed format) the steps required to perform garbage collection. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3   In the terminal window, at the command line prompt, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1   - Science Software Integration and Test (SSI&T)
  - TS2   - New Version Checkout
  - OPS   - Normal Operations
- "utilities" is the directory containing the production processing utility/start-up scripts (e.g., EcDpPrAutosysStart).
- Note that the separate subdirectories under /usr/ecs apply to (describe) different operating modes.

4   Start the garbage collection script by entering:

EcDpPrGarbageCollectorStart <MODE> <machine> <retries> <interval>

- **<machine>** refers to the name of the virtual computer (e.g., x0spg11_vc) for which garbage collection is being requested.
  - The value entered is assigned to a MACHINE_TO_COLLECT variable.
- **<retries>** is the number of retries in case of database deadlock.
  - The default number of retries is five (5) [times].
- **<interval>** is the amount of time (in minutes) between retries.
  - The default interval is 30 [minutes].

5   Observe the results as the script runs.
- A UNIX command line prompt is displayed when the script has finished.

Table 13.5-3. Perform Garbage Collection - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server host)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcDpPrGarbageCollectorStart <MODE> <machine> <retries> <interval>**	**enter text, press Enter**
5	Observe script results	**read text**

13.5.3 Run the Deletion Server Client

In the /usr/ecs/*MODE*/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrDeletionClientStart) that can be run to delete unneeded files from the science processing disks and update the PDPS database accordingly. The executable invoked by the script is EcDpPrDeletionClient.

The script for running the Deletion Server Client is invoked from a UNIX command line prompt. Table 13.5-4 presents (in a condensed format) the steps required to run the Deletion Server Client. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   Set the application environment variables by entering:

     setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3   In the terminal window, at the command line prompt, enter:

     cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1   - Science Software Integration and Test (SSI&T)
  - TS2   - New Version Checkout
  - OPS   - Normal Operations
- "utilities" is the directory containing the production processing utility/start-up scripts (e.g., EcDpPrAutosysStart).

- Note that the separate subdirectories under /usr/ecs apply to (describe) different operating modes.

4   Start the Deletion Server Client by entering:

EcDpPrDeletionClientStart <MODE>

5   Observe the results as the script runs.
- A UNIX command line prompt is displayed when the script has finished.

Table 13.5-4. Run the Deletion Server Client - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server host)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcDpPrDeletionClientStart <MODE>**	**enter text, press Enter**
5	Observe script results	**read text**

13.5.4 Resolve PDPS Database and Science Processing Disk Content Discrepancies

In the /usr/ecs/<MODE>/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrRmFilesWOGranules.pl) that can be run to ensure consistency between the file references in various tables in the PDPS database and the files actually staged on the science processing disks. When it runs, the script performs the following functions:

- Generates a list of files with consistent references among various tables in the PDPS database for the specified mode.
- Checks for files on the disk that are not included in the list of files referenced in the PDPS database and either lists the inconsistent files or generates a script to delete them (as specified by the person running the script).
- Checks to determine whether the disk partitions referenced in the PDPS database actually exist on the disk(s).
  − If the disk partitions are not on the disk(s), the script removes references to them from the database.
- Removes all file references in the PDPS database that are not included in the list of files.
  − The script resets the PlRscDiskPartition according to the remaining DpPrDiskPartition entries (if specified by the person running the script).

The script for resolving PDPS database and science processing disk content discrepancies is invoked from a UNIX command line prompt. Table 13.5-5 presents (in a condensed format) the steps required to resolve PDPS database and science processing disk content discrepancies. If

you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

NOTE: The **EcDpPrRmFilesWOGranules.pl** script should be run when the system is relatively quiet; i.e., when no jobs are running in AutoSys.

1   Access a terminal window logged in to the Planning/Management Workstation host.
   - Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
   - For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   In the terminal window, at the command line, enter:

   cd /usr/ecs/<MODE>/CUSTOM/utilities

   - **<MODE>** is current mode of operation.
     − TS1   - Science Software Integration and Test (SSI&T)
     − TS2   - New Version Checkout
     − OPS   - Normal Operations
   - "utilities" is the directory containing the Planning Subsystem utility/start-up scripts (e.g., EcDpPrAutosysStart).

3   Start the database cleaning script by entering:

   EcDpPrRmFilesWOGranules.pl <database user> <database password> <MODE> <database server> fix [or **nofix**]

   - **<database user>** is the user name for logging in to isql.
   - **<database password>** is the password for isql login.
   - **<database server>** refers to the name of the PDPS database server (e.g., x0sps02_srvr).
   - **fix** (or **nofix**) specifies whether or not inconsistent files are removed from the science processing disks.
   - An option is to add **>& fix.log** or **>& /usr/ecs/<MODE>/CUSTOM/logs/fix.log** to the end of the command line to direct the output of the script to a log file.
   - For example:

   EcDpPrRmFilesWOGranules.pl pdps_role password1 OPS x0sps02_srvr nofix

4   Observe the results as the script runs.
   - A UNIX command line prompt is displayed when the script has finished.

Table 13.5-5. Resolve PDPS Database and Science Processing Disk Content Discrepancies - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	**EcDpPrRmFilesWOGranules.pl <database user> <database password> <MODE> <database server> fix** [or **nofix**]	**enter text, press Enter**
4	Observe script results	**read text**

13.5.5 Save and/or Reset the PDPS Database

The utilities directory (/usr/ecs/<MODE>/CUSTOM/utilities) contains the following three scripts that are used in resetting the database:

- EcPlDbReset.
- EcPlDbList.
- EcPlDbSave.

Saving the database using the EcPlDbSave script produces one ASCII file (with a ".dat" extension) for each database table and writes each file to a specified directory. The saved data can be restored to the database by running the EcPlDbReset script with the name of the saved database as an argument.

Resetting the database involves clearing ("wiping out") the data in the database tables and loading values from a specified "saved database" file. It is important to take into consideration the consequences of resetting the database before performing the procedure. Resetting the database removes and replaces **all** Resource Definitions, Resource Reservations (Ground Events), Production Requests, Data Processing Requests, and Production Plans. There should be coordination with all affected parties, including the Resource Planner, Production Planner, and Production Monitors concerning the effects of resetting the database as well as its after-effects (e.g., recreating resource definitions, resource reservations, and production requests).

Whenever the PDPS database is reset (not including database cleanup by running the EcPlDbClean script) it is also necessary to remove all PLS subscriptions in the Communications Subsystem (CSS) Subscription Server database (where the subscriber is Subscription Manager). Production personnel can remove the subscriptions using the Subscription Server GUI (EcSbSubServerGUI) if they have access to the GUI. Otherwise, they can request User Services personnel to remove the subscriptions.

As a result of removing the subscriptions, no subscription notification will come through for existing jobs in the newly loaded database. Only new jobs generated using the Production Request Editor will work normally with regard to subscriptions.

The scripts used in saving and/or resetting the database are invoked from a UNIX command line prompt. Table 13.5-6 presents (in a condensed format) the steps required to save and/or reset the PDPS database. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

NOTE: It is important to log in as a user who has "write" permission in the saved_dumps directory (/usr/ecs/<MODE>/CUSTOM/utilities/saved_db/saved_dumps); otherwise it will not be possible to save database contents.

1 Access a terminal window logged in to the Planning/Management Workstation host.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem utility/start-up scripts.

4 If saving the current database is desired, start the process of saving the database by entering:

EcPlDbSave <Saved_DB_Name>

- **<Saved_DB_Name>** is the desired name for the directory into which the data files will be written.
- If there is not already a directory with the specified **<Saved_DB_Name>**, a **"Making directory..."** message is displayed.
- If there is already a directory with the specified **<Saved_DB_Name>**, the following prompt is displayed:

  Are you sure you wish you overwrite previously saved version of <Saved_DB_Name>? (Y/N)

5   If the prompt **Are you sure you wish you overwrite previously saved version of <Saved_DB_Name>? (Y/N)** appears, enter either **Y** or **N** (as appropriate)

- If **y** is entered, the data are saved in the specified directory, replacing the data that had previously been saved there.
- If **n** is entered, an **"Exiting without saving database..."** message is displayed.

6   If obtaining a listing of saved databases is desired, generate a listing of the saved databases by entering:

EcPlDbList

- A listing of saved databases is displayed.
- The listing is a useful tool to display the options for resetting the database.

7   If resetting the current database is desired, start the database reset process by entering:

EcPlDbReset <Saved_DB_Name>

- **<Saved_DB_Name>** refers to the name of the directory containing the data files to be loaded into the database.
- The data in the database are replaced with the data from the specified directory.
- When the database has been reset a **"Successfully reset the database to <Saved_DB_Name>"** message is displayed.

Table 13.5-6. Save and/or Reset the PDPS Database - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcPlDbSave <Saved_DB_Name>** (if applicable)	**enter text, press Enter**
5	Either **Y** or **N** (if applicable)	**enter text, press Enter**
6	**EcPlDbList** (if applicable)	**enter text, press Enter**
7	**EcPlDbReset <Saved_DB_Name>** (if applicable)	**enter text, press Enter**

13.6 Tuning System Parameters

The Production Planner works with the rest of the production team (i.e., Resource Planner and Production Monitors) to tune system parameters. Detailed procedures are included in Chapter 14, Production Processing.

13.7 Troubleshooting Production Planning Problems

Troubleshooting is a process of identifying the source of problems on the basis of observed trouble symptoms. One common source of problems involves connections with other subsystems for the transmission of messages or data. Like many other operational areas in the system, Planning has interfaces with many other subsystems. Consequently, problems with processing can be traced to either the Planning Subsystem or one of many other subsystems, including (but not necessarily limited to) those in the following list:

- Data Processing Subsystem (DPS).
- Data Server Subsystem (DSS).
- Communications Subsystem (CSS).

Table 13.7-1, below, provides an Activity Checklist for troubleshooting Production Planning problems.

Table 13.7-1. Troubleshooting Production Planning Problems - Activity Checklist (1 of 2)

Order	Role	Task	Section	Complete?
1	Production Planner	Troubleshoot a Production Planning Problem	(P) 13.7.1	
2	Production Planner	Check Connections to Hosts/Servers	(P) 13.7.1.1	
3	Production Planner	Check Log Files	(P) 13.7.1.2	
4	Production Planner	Check Database Connections	(P) 13.7.1.3	
5	Production Planner	Handle a Failure to Generate DPRs	(P) 13.7.2	
6	Production Planner	Check the Production Request Editor ALOG File	(P) 13.7.2.1	
7	Production Planner	Use ISQL to Check Database Tables	(P) 13.7.2.2	
8	Production Planner	Check the PDPS Database for Causes of Failure to Generate DPRs	(P) 13.7.2.3	
9	Production Planner	Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data	(P) 13.7.2.4	
10	Production Planner	Check the Production Request Editor Debug File for Evidence of Metadata Queries	(P) 13.7.2.5	
11	Production Planner	Respond to PR or DPR Deletion that Hangs	(P) 13.7.3	
12	Production Planner	Check for Database Deadlocks	(P) 13.7.3.1	
13	Production Planner	Check for Resource Locks in the PDPS Database	(P) 13.7.3.2	
14	Production Planner	Respond to DPR Deletion that Fails	(P) 13.7.4	
15	Production Planner	Handle a DPR Scheduling Failure	(P) 13.7.5	
16	Production Planner	Respond to a "DPR Validation Failed" Error	(P) 13.7.5.1	

Order	Role	Task	Section	Complete?
17	Production Planner	Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error	(P) 13.7.5.2	

Fault Recovery

Each request that crosses a client/server boundary is assigned a system-unique identifier referred to as an RPC ID. (RPC refers to Remote Procedure Call, the mechanism by which requests are submitted from client to server.) The RPC ID facilitates the automatic fault recovery events that occur whenever there is a client or server failure.

- As a request propagates through the system, each associated client/server exchange is assigned a unique RPC ID.
    - The RPC ID for each interaction is derived from the previous RPC ID received by the client for the request. Consequently, all RPC IDs associated with a given request have a common portion that relates the various client/server calls to one another.
    - Given the previous RPC ID, clients consistently reproduce the same RPC ID that was submitted to the server on the subsequent event.
- The concept of reproducible RPC IDs is central to the system's fault recovery capability.
    - When requests are retried from client to server, they are always submitted with the same RPC ID that was used in the original submission of the request, even if either the client or server has crashed between retries.
- The RPC ID is also central to the check-pointing aspect of fault recovery.
    - As requests arrive at fault recovery-enabled servers, they are recorded in a persistent store (typically a database), tagged with the RPC ID, which identifies the request.
    - As the request is serviced, check-pointing state information may be updated in the persistent store, up to and including the completion status of the request.
    - This allows the servers to resume servicing from the last check-pointed state, particularly upon resubmission from a client.

PLANG and PRONG components check-point the following types of information:

- **EcPlOdMgr** - Request information.
- **EcPlSubMgr** - Unprocessed subscription notifications.
- **EcDpPrDeletion** - Interim Delete Requests.
- **EcDpPrEM** - Queued and activated jobs.

Fault Handling

Failure events are classified according to the following three severity levels:

- Fatal error.
  - Returned when a request cannot be serviced, even with operator intervention.
  - For example, if a request is made to distribute data via ftp to a non-existent host, the request is failed with a fatal error.
- Retry error.
  - Potentially recoverable error.
  - Normally, a retry error would be returned to the client only when the server cannot recover from the error automatically.
  - A retry error may require operator assistance during recovery. For example, when using the Production Request Editor GUI, the Production Planner would enter a new name for a production request after being notified that a previously entered name contained too many characters.
- Warning.
  - Provided when operations can proceed without interruption, but an unexpected circumstance was detected.
  - For example, if a client requests removal of a file but the file does not exist, there is no error per se; however, a warning is generated to caution the client that the file to be removed did not exist in the first place.

Transient errors (such as network errors) are always retry errors.

- In general, clients and servers that experience transient retry errors first attempt to recover by retrying the operation automatically.
- One special case of this is "rebinding," which refers to the process by which a client automatically attempts to re-establish communication with a server in the event communication is disrupted.
  - The disruption may be caused by transient network failure, or by the server crashing or being brought down.
  - In any case, the client automatically attempts to reconnect to the server for a configurable period of time on a client-by-client basis.

Processes encountering an error or receiving an error from a server request can either pass the error back to a higher-level client or present it to the operator for operator intervention. The specific fault handling policies for PLANG and PRONG client processes are shown in Table 13.7-2.

Table 13.7-2. PLANG and PRONG Fault Handling Policies

Client Process	Fault Handling Policy
EcPlSubMgr	**Retry errors:** All Subscription processing errors are retried a configurable number of times and for a configurable time period. After the configurable number of times (or time period) the subscription is lost. **Fatal errors:** N/A.
EcPlPREditor_IF **EcPlWb**	**Retry errors:** Since these are GUI applications, errors are reported to the operator and it is the operator's responsibility to retry the request. **Fatal errors:** Errors are reported to the operator.
EcPlOdMgr	**Retry errors:** Retries errors from the Science Data Server and the Subscription Server. **Fatal errors:** Logs errors and stops current on demand requests.
EcDpPrEM	**Retry errors:** Errors are retried a configurable number of times, then the job is failed and it is up to the Production Monitor to restart the job through AutoSys. **Fatal errors:** A fatal error message is logged.
EcDpPrJobMgmt	**Retry errors:** If a DPR cannot be assigned to a machine or created in AutoSys, it is left in a PENDING state and the assignment is retried after DpPrPendingThreadWaitInterval seconds. **Fatal errors:** N/A.
EcDpPrDeletion	**Retry errors:** No retries are implemented. Status from DSS is <u>not</u> checked. **Fatal errors:** N/A.

Client Crash and Restart

When a client of a PLANG or PRONG server crashes, the server (i.e., EcPlSubMgr, EcDpPrJobMgmt, or EcDpPrDeletion) continues to service the requests that were in process at the time of the client's crash.

When a client restarts in the system, it sends a restart notification to each server with which it interacts.

- Clients notify servers that they have come up either "cold" or "warm."
- Generally, the notification temperature sent to the server matches the temperature at which the client process is restarted.

The default server behavior in response to startup notification from a client is as follows:

- Warm Notification.
    - Outstanding requests for the restarted clients remain available in the persistent store.
    - The outstanding requests may be resubmitted by the client, and are serviced to completion upon resubmission.
    - Associated resources are left allocated until the requests are completed.

- Cold Notification.
  - All outstanding requests for the restarted client are cancelled.
  - If the client resubmits any cancelled request using the same RPC ID (e.g., by pressing the Retry button from an operator GUI), it is failed with a fatal error due to the client cold startup notification.
  - Any resources associated with the cancelled requests are released and reclaimed by the system.

Server Crash and Restart

When a server crashes, clients cannot continue to submit requests for processing.

- Synchronous requests in progress result in a Distributed Computing Environment (DCE) exception being thrown back to the client process, which enters a rebinding failure recovery mode (as previously mentioned).
- Attempts to submit requests while the server is down result in the client blocking until a communication timeout has been reached.
- Although DCE has been replaced by socket-based library calls (i.e., CCS Middleware), the DCE exception code is handled by the CCS Middleware.

When a server restarts, it may perform various resynchronization activities in order to recover from an unexpected termination.

- In the event of a server cold start or cold restart, the server typically cancels all outstanding requests and reclaims all associated resources.
- In general, existing request queues are retained for warm restarts and cleared for cold starts or cold restarts.
- **EcPlOdMgr**-specific activities upon start/restart:
  - **Warm Restart:** Any requests that have not been processed are read from the database and processed.
    - Cold Start or Cold Restart: N/A.
- **EcPlSubMgr**-specific activities upon start/restart:
  - **Warm Restart:** Any subscriptions that have not been processed are read from checkpoint file and processed.
  - **Cold Start or Cold Restart:** N/A.
- **EcDpPrDeletion**-specific activities upon start/restart:
  - **Warm Restart:** Interim granules marked for deletion are read from the database and are deleted when time-out occurs.
    - Cold Start or Cold Restart: N/A.
- **EcDpPrJobMgmt**-specific activities upon start/restart:
  - **Warm Restart:** Jobs in AutoSys and jobs waiting in the queue are read from the database; any jobs that are ready are placed into AutoSys from the queue (if there are processing slots available).
  - **Cold Start or Cold Restart:** N/A.

Request Resubmission

Upon restarting a crashed client or server, requests are typically resubmitted. If the restarted process was started warm, the fault-recovery capabilities permit the server to resume processing of the request from its last check-pointed state. This prevents needless repetition of potentially time-consuming activities.

- **EcDpPrJobMgmt**- and **EcDpPrDeletion**-specific activities upon resubmission of a request:
  - Requests are submitted synchronously.
  - If the entire request is resubmitted by a client, then only that part of the resubmitted request that has not been completed is reprocessed.

13.7.1 Troubleshoot a Production Planning Problem

1   If it is not possible to log in to the Planning Subsystem host, ask the Operations Controller/System Administrator to verify that the host is "up."
- Examples of Planning Subsystem host names include **e0pls03**, **g0pls01**, and **l0pls02**.

2   If the GUI (e.g., the **Production Request Editor**, the **Production Strategies GUI**, or the **Planning Workbench**) is not displayed when the start-up script has been properly invoked, ensure that the DISPLAY variable was set properly.
- For detailed instructions refer to the applicable procedure.
  - **Launch the Production Request Editor** (Section 13.2.2).
  - **Launch the Production Strategies GUI** (Section 13.3.1).
  - **Launch the Planning Workbench and Planning Timeline GUIs** (Section 13.4.1).

3   If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in a different mode from the person using the selected Application ID, use a different Application ID.
- For detailed instructions refer to the applicable procedure.
  - **Launch the Production Request Editor** (Section 13.2.2).
  - **Launch the Planning Workbench and Planning Timeline GUIs** (Section 13.4.1).

4   If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in the same mode as the person using the selected Application ID, coordinate use of Planning applications with the other user and/or the System Administrator.

5   If an error message associated with the Production Request Editor is received, refer to Table 13.7-3, Production Request Editor User Messages.
- The table is adapted from the corresponding table in 609-EMD-001, *Release 7.10 Operations Tools Manual for the EMD Project*.

6 If an error message associated with the Production Strategies GUI is received, refer to Table 13.7-4, Production Strategies GUI User Messages.
- The table is adapted from the corresponding table in 609-EMD-001, *Release 7.10 Operations Tools Manual for the EMD Project*.

7 If an error message associated with the Planning Workbench is received, refer to Table 13.7-5, Planning Workbench User Messages.
- The table is adapted from the corresponding table in 609-EMD-001, *Release 7.10 Operations Tools Manual for the EMD Project*.

8 If a Production Request fails (DPR generation fails), ensure that it is possible to connect to the necessary hosts and servers.
- For detailed instructions refer to the **Check Connections to Hosts/Servers** procedure (Section 13.7.1.1).

10 If a Production Request fails (DPR generation fails) and if the servers are all "up," perform the **Handle a Failure to Generate DPRs** procedure (Section 13.7.2).

11 If a PR or DPR deletion hangs, ensure that enough time has passed to allow DPR deletion.
- Deleting a DPR can require as much time as creating a DPR.

12 If a PR or DPR deletion hangs and enough time has passed to allow DPR deletion, check for a database lock or resource lock in the PDPS database.
- For detailed instructions refer to the **Respond to PR or DPR Deletion that Hangs** procedure (Section 13.7.3).

13 If DPR deletion fails, ensure that enough time has passed to allow DPR deletion.
- Deleting a DPR can require as much time as creating a DPR.

14 If DPR deletion fails and enough time has passed to allow DPR deletion, check the Deletion Server Debug log (EcDpPrDeletionDebug.log).
- For detailed instructions refer to the **Respond to DPR Deletion that Fails** procedure (Section 13.7.4).

15 If DPR scheduling fails (DPR is not passed to Data Processing), perform the **Handle a DPR Scheduling Failure** procedure (Section 13.7.5).

16 If some other type of problem is encountered, check the log files for error messages.
- Examples of log files include EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, and EcPlTl.ALOG.
- Log files are located in the /usr/ecs/<MODE>/CUSTOM/logs directory.
- For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).

17 If the problem cannot be identified and fixed without help within a reasonable period of time, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-3. Production Request Editor User Messages (1 of 5)

Message Text	Impact	Cause and Corrective Action
Data Processing Request not open, Data Processing Request to be open must be selected.	Unable to open a DPR.	1. **Single-click** on a DPR from the DPR List. 2. Open the DPR using either **File → Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]
Do you want to delete this DPR <DPRID>?	Requires operator input (confirmation of deletion) before proceeding.	1. **Single-click** on **Yes** to delete the DPR. 2. **Single-click** on **No** to keep the DPR.
Do you want to delete this PR <PRNAME>?	Requires operator input (confirmation of deletion) before proceeding.	1. **Single-click** on **Yes** to delete the PR. 2. **Single-click** on **No** to keep the PR.
DPR Delete Failed.	Selected DPR was not deleted from the PDPS database.	Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Dpr Generation Incomplete for PR <PRNAME>. Do you want to complete Dpr explosion?	Requires operator input before proceeding.	1. **Single-click** on **Yes** to complete the DPR generation. 2. **Single-click** on **No** to display another message that will show how many DPRs have been generated so far.
Environment variable PL_NEW not set.	**Production Request Editor** GUI is not started	1. Submit a request to the Database Administrator to check the value assigned to **PL_NEW** (associated with EcPlPREditor) in the Configuration Registry. 2. If no value is specified for **PL_NEW**, submit a request to the Database Administrator to add the value **PL_NEW = New** for the EcPlPREditor in the Configuration Registry.
Invalid Time is entered or End time is less than Begin Time.	Unable to create Data Processing Requests because the collection/insertion time(s) entered are invalid.	Ensure that the **Begin** and **End** dates/times entered on the **PR Edit** screen are valid.
Must select a PGE for Production Request before saving.	Unable to proceed with Data Processing Request creation until a PGE is entered on the **PR Edit** screen.	1. Open the **PGE** screen (**single-click** the mouse on the **PGE** button of the **PR Edit** screen). 2. Select (highlight) a PGE from the PGE list. 3. **Single-click** on the **OK** button.

Table 13.7-3. Production Request Editor User Messages (2 of 5)

Message Text	Impact	Cause and Corrective Action
\<number\> Dpr(s) have previously been generated. Are you sure you want to delete this PR?	Requires operator input before proceeding.	1. **Single-click** on **Yes** to delete the PR. 2. **Single-click** on **No** to clear the message.
Please select a Dpr you want to delete from the DPR List.	DPR will not be deleted from the PDPS database until a DPR is selected from DPR List.	1. **Single-click** on a DPR from the DPR List. 2. Delete the DPR using either **Edit → Delete** from the pull-down menu or the keyboard **Ctrl-D**. [For detailed instructions refer to the **Delete a Data Processing Request** procedure (Section 13.2.8).]
Please select a DPR you want to open from the DPR list,	DPR cannot be opened on the DPR View page until the DPR is selected from the DPR List.	1. **Single-click** on a DPR from the DPR List. 2. Open the DPR using either **File → Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]
Please select a PR you want to delete from the PR List.	Cannot delete a Production Request until it is selected from the PR List.	1. **Single-click** on a PR from the PR List. 2. Delete the PR using either **Edit → Delete** from the pull-down menu or the keyboard **Ctrl-D**. [For detailed instructions refer to the **Delete a Production Request** procedure (Section 13.2.6).]
Please specify production request to filter by.	Unable to filter Production Request IDs.	**Filter** button acts on the pattern specified in the filter text entry area. Here, the user can filter all the production requests by the specified pattern. 1. If applicable, **single-click** on the appropriate PR type on the **PR Type** option button. 2. **Single-click** on the appropriate PR in the **PR Name** field. 3. **Single-click** on the **Filter** button. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]
PR Failed - Need to include 1st orbit in the Orbit model.	Data Processing Request creation fails because 1st orbit information is incorrect.	1. Enter isql commands for checking the PDPS database PlPgeOrbitModel table to determine whether the orbit information for the first orbit is in the table. [For detailed instructions refer to the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).] 2. If the orbit information for the first orbit is not in the PlPgeOrbitModel table, notify the SSI&T team.
PR Failed - Read failure from PlOrbitModel.	Data Processing Request creation fails because orbit information cannot be read from the PDPS database.	Unable to read the PlPgeOrbitModel table from the PDPS database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-3. Production Request Editor User Messages (3 of 5)

Message Text	Impact	Cause and Corrective Action
Production Request <PRNAME> Deleted; <number> DPR(s) deleted from the database.	Production Request(s) and dependent DPR(s) are successfully deleted from the PDPS database.	For information only. The message indicates how many DPRs associated with the PR were deleted.
Production Request Delete Failed.	Selected Production Request(s) is (are) not deleted.	Unable to delete a PR from the database. Check the log files (e.g., EcPIPREditor.ALOG and EcPIPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Production Request Explosion into DPR(s) Failed.	Data Processing Request creation failed.	Check the log files (e.g., EcPIPREditor.ALOG and EcPIPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Production Request Explosion into DPRs Failed, zero DPRs Generated.	Data Processing Request creation failed.	Check the log files (e.g., EcPIPREditor.ALOG and EcPIPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Production Request Explosion into DPRs ok. <number> DPR(s) Generated.	Data Processing Requests successfully created and stored in the PDPS database.	For information only. The message indicates how many DPR(s) were generated during the PR Explosion.
Production Request Explosion into DPR(s) ok. <number> more DPR(s) Generated.	Data Processing Requests successfully created and stored in the PDPS database.	For information only. The message indicates how many more DPRs were generated for the PR.
Production Request not open, Production Request to be open must be selected.	Production Request is not opened on the **PR Edit** screen until a PR is selected from the PR List.	1. **Single-click** on a PR from the PR List. 2. Open the PR using either **File → Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]
Production Request not saved, Production Request already exists.	Data Processing Request creation failed because the Production Request name already exists in the PDPS database.	1. Enter a new name for the production request. 2. Save the PR. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]

Table 13.7-3. Production Request Editor User Messages (4 of 5)

Message Text	Impact	Cause and Corrective Action
Production Request not saved, Production Request must have a name before it is saved.	Unable to proceed with Data Processing Request creation because no name has been entered for the Production Request.	1. Enter a new name for the production request. 2. Save the PR. [For detailed instructions refer to the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).]
Production Request not saved, Production Request must have a name shorter than 20 characters before it is saved	Unable to proceed with Data Processing Request creation because the name chosen for the Production Request exceeds the maximum number of characters allowed.	1. Enter a PR name with fewer than 20 characters. 2. Save the PR. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]
SECURITY VIOLATION: no write permission.	Unable to proceed with Data Processing Request creation because Production Request Editor does not have proper "write" permission on the PDPS database.	Operator does not have permission to save a production request. 1. Exit from the Production Request Editor (**File → Exit** from the pull-down menu). 2. Log in as a user with "write" permission. [Contact the System Administrator for assistance if necessary.] 3. Launch the Production Request Editor. [For detailed instructions refer to the **Launch the Production Request Editor** procedure (Section 13.2.2).] 4. Create the Production Request. [For detailed instructions refer to the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).]
Unable to Initialize PlDpr Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the DPR table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Unable to Initialize PlPge Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the PGE table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-3. Production Request Editor User Messages (5 of 5)

Message Text	Impact	Cause and Corrective Action
Unable to Initialize PlProductionRequest Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the PR table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Write to Database of Production Request Failed.	Data Processing Request creation fails because the Production Request information cannot be saved in the PDPS database.	Unable to write the data to the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Zero DPR Generated. Production Request Not Saved.	Data Processing Request creation failed.	The proposed Production Request might result in the creation of a duplicate DPR. 1. Compare the characteristics of the proposed PR and existing DPRs to determine whether the Production Request would result in the creation of a duplicate DPR. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).] 2. Ensure that it is possible to connect to the necessary hosts and servers (listed in Table 13.7-7). [For detailed instructions refer to the **Check Connections To Hosts/Servers** procedure (Section 13.7.1.1).] 3. If hosts/servers are all "up," perform the **Handle a Failure to Generate DPRs** procedure (Section 13.7.2). 4. Retry generating DPRs by resaving the Production Request. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).] 5. Check the log files (e.g., EcPIPREditor.ALOG and EcPIPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]

Table 13.7-4. Production Strategies GUI User Messages

Message Text	Impact	Cause and Corrective Action
Invalid StrategyID. Please enter a StrategyID.	Unable to save the Production Strategy.	The strategy ID text field is blank. Enter a valid Strategy ID. [For detailed instructions refer to the **Define or Modify a Production Strategy** procedure (Section 13.3.2).]
No StrategyID selected. Please select one.	Unable to open or delete a Strategy	No strategy is selected from the Production Strategy list. **Single-click** on a Strategy ID from the list.
This StrategyID already exists.	Unable to save Production Strategy.	The Strategy ID entered already exists. Enter a new name for the Strategy ID.
Total Weight must be 100 – Normalize weights.	Unable to save Production Strategy.	The combined weights for the different "Types" exceed 100. **Single-click** on the **Normalize** push button.
Value out of range (0-10).	Unable to save the Production Strategy.	Default value entered for the selected "Type" is out of range and is reset to this strategy's default value for this "Type". Enter a value in the range of 0 – 10.
Value out of range (0-100).	Unable to save the Production Strategy.	Weight for the selected "Type" is out of range and is reset to this strategy's default weight for this "Type". Enter a value in the range of 0 – 100.

Table 13.7-5. Planning Workbench User Messages (1 of 2)

Message Text	Impact	Cause and Corrective Action
The following dpr jobs were failed when you activated the plan <plan name> <list of failed dprs, one per line> Do you want to recover the failed dpr jobs?	Planning Workbench detects that Job Management has not received the DPRs correctly. The operator needs to be aware that there is (are) failed job(s).	**Single-click** on **Recover**; the failed jobs will be resubmitted.
You cannot reschedule with the current active plan. Do you want to create new plan to schedule production request(s) and to reactivate the plan?	Cannot replan under the same plan name.	1. **Single-click** on **Yes** to create a new plan that can be used for replanning. 2. **Single-click** on **No** and use an existing plan to activate and replan.

Table 13.7-5. Planning Workbench User Messages (2 of 2)

Message Text	Impact	Cause and Corrective Action
You have already submitted the plan, <plan name> and this plan is currently active. Do you want to create a new plan?	If the operator has already submitted this plan for activation, the Planning Workbench will not allow it to be submitted twice.	1. **Single-click Yes** to create a new plan that can be used for replanning. 2. **Single-click No** and use an existing plan to activate and replan.
You need to save the current plan before you activate the plan!!!	Insures consistency between the database active plan and what actually gets activated.	**Single-click** on **Yes** to save the plan. The next time activation is attempted there will be no error message.

13.7.1.1   Check Connections to Hosts/Servers

The procedure to **Check Connections to Hosts/Servers** is a part of the **Troubleshoot a Production Planning Problem** procedure (Section 13.7.1).   Table 13.7-6 presents (in a condensed format) the steps required to check connections to hosts/servers. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1   Access a terminal window logged in to the Planning/Management Workstation.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- Most other system hosts are acceptable for checking connections.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- Change directory to the directory containing the utility scripts.

3   At the command line prompt enter:

EcCsIdPingServers <MODE>

- The following type of response is displayed (only a few representative lines are shown):

   /usr/ecs/TS2/CUSTOM/bin/CSS/Sweeper -nsh x0icg01 -nsp 18202
   FoSwSweeper application started...
   We made a connection with EntryId =x0ins01:38709:23057 ---
   EcSrTransportSubServer

**We made a connection with EntryId =x0ins01:38712:23057 ---
EcSrTransportSubEventServer
We made a connection with EntryId =x0acs06:33379:17033 --- DsShQuitIDL
We made a connection with EntryId =x0wkg01:11959:41838305 ---
EcDsHdfEosServer_3_G3
[…]**

4   Observe the results displayed on the screen to determine whether connections can be made with the necessary hosts and servers.

- The necessary hosts and servers are listed in Table 13.7-7, Hosts, Servers, Clients and Other Software Relevant to Production Planning and Processing.

5   If pinging the servers (Step 3) indicated a problem with any connection, ping the servers again (at the command line prompt enter: **EcCsIdPingServers <MODE>**).

6   Observe the results displayed on the screen to determine whether connections can be made with the necessary hosts and servers.

7   If it is not possible to connect to any needed host(s)/server(s), notify the Operations Controller/System Administrator to check the hosts/servers and bring them back up if necessary.

8   Return to the procedure that recommended checking connections to hosts.

Table 13.7-6. Check Connections to Hosts/Servers - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	**EcCsIdPingServers <MODE>**	**enter text, press Enter**
4	**EcCsIdPingServers <MODE>** [again]	**enter text, press Enter**
5	Identify hosts and servers with which connections cannot be made	**read text**
6	Notify the Operations Controller/System Administrator to bring hosts/servers back up (if applicable)	**contact Operations Controller**
7	Return to the procedure that recommended checking connections to hosts	

Table 13.7-7. Hosts, Servers, Clients and Other Software Relevant to Production Planning and Processing

HOST	SERVER/CLIENT/OTHER SOFTWARE
Planning/Management Workstation	Production Request Editor (EcPlPREditor)
	Planning Workbench GUI (EcPlWb)
	Production Strategies GUI (EcPlProdStrat)
	Production Planning Master Timeline (EcPlTl)
	Message Handler (EcPlMsh)
	System Name Server (EcPlSns)
	Resource Model (EcPlRm)
Queuing Server (e.g., x0sps04)	Job Management Server (EcDpPrJobMgmt)
	Deletion Server (EcDpPrDeletion)
	Execution Management (EcDpPrEM)
	AutoSys Event Processor (event_demon)
	AutoSys Event Server (Sybase server) (e.g., x0sps02_srvr)
	On-Demand Manager (EcPlOdMgr)
	Subscription Manager (EcPlSubMgr)
	PDPS database Sybase server (e.g., x0sps02_srvr)
Science Processor (e.g., x0spg11)	PGE Management (EcDpPrRunPGE)
	Resource Usage (EcDpPrRusage)
	PGEs
Access/Process Coordinators (APC) Server (e.g., x0acg01)	Archive Server (EcDsStArchiveServer)
	FTP Server (EcDsStFtpServer)
	Cache Manager Server (EcDsStCacheManagerServer)
	Staging Disk Server (EcDsStStagingDiskServer)
	Pull Monitor Server (EcDsStPullMonitorServer)
Ingest Server (e.g., x0icg01)	Name Server (EcCsIdNameServer)
	Registry Server (EcCsRegistry)
Sun Consolidation External Server (e.g., x0ins01)	Data Dictionary (EcDmDictServer)
Sun Consolidation Internal Server (e.g., x0acs11)	Science Data Server (EcDsScienceDataServer)
	Subscription Server (EcSbSubServer)

NOTE: Depending on the installation, software may be loaded on hosts other than the examples provided.

13.7.1.2   Check Log Files

Log files can provide indications of the following types of problems:

- Communication problems.
- Database problems.
- Lack of disk space.

Table 13.7-8 presents (in a condensed format) the steps required to check log files. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the appropriate host.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- **<MODE>** is current mode of operation.
    - TS1   - Science Software Integration and Test (SSI&T)
    - TS2   - New Version Checkout
    - OPS   - Normal Operations
- "logs" is the directory containing Production Planning log files (e.g., EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, and EcPlTl.ALOG).

3   At the command line prompt enter:

pg <file name>

- **<file name>** refers to the Production Planning log file to be reviewed (e.g., EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, or EcPlTl.ALOG).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4   Review the log file to identify problems that have occurred.
- To exit from **pg** at the **:** prompt enter:

    q
    - The command line prompt is displayed.

5   Respond to problems as follows:
- Production Planning-related problems.
    - Perform the appropriate procedure(s) from Table 13.7-1, Troubleshooting Production Planning Problems.
- Communication problems.
    - Notify the Operations Controller/System Administrator of suspected communication problems.
- Database problems.
    - Verify that relevant database servers are running.

- Check for lack of (or corruption of) data in the database using either a database browser or interactive structured query language (isql) commands.
- Notify the Database Administrator of suspected database problems.
- Lack of disk space.
  - Remove unnecessary files.
  - Notify the Operations Controller/System Administrator of recurring disk space problems.

Table 13.7-8. Check Log Files - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Identify problems indicated in the log file	**read text**
5	Respond to problems as necessary	

13.7.1.3   Check Database Connections

If applications (including the GUIs) are unable to connect to the database, data cannot be retrieved or (in the case of the GUIs) displayed. Consequently, if a GUI does not display data or if the display does not refresh, checking the database connections is a logical step in trying to isolate the problem.

Table 13.7-9 presents (in a condensed format) the steps required to check database connections. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Submit a request to the Database Administrator to identify the values for parameters associated with the appropriate application.
- The following parameters should be requested:
  - **DBName.**
  - **DBServer.**
  - **DBMaxConnections.**
- The preceding parameters are associated with the following applications:
  - EcPlPREditor.
  - EcPlProdStrat.
  - EcPlRm.
  - EcPlSubsEdit.
  - EcPlTl.
  - EcPlWb.

2   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

3   At the command line prompt enter:

isql –U <User ID> -S <Database Server>

- **<User ID>** is the database user's identification; e.g., **pdps_role**.
- **<Database Server>** is the database server; e.g., **x0sps02_srvr**.
- For example:

    isql –U pdps_role –S x0sps02_srvr

4   At the **Password:** prompt enter:

<database password>

- **<database password>** is the password for logging in to the database using the specified **<User ID>**.
- A **1>** prompt is displayed, indicating that a connection has been made with the database.

5   At the **1>** prompt enter:

sp_who

6   At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns::
  - **spid.**
  - **status.**
  - **loginame.**
  - **hostname.**
  - **blk.**
  - **dbname.**
  - **cmd.**

7   At the **1>** prompt enter:

sp_configure "user connections"

8   At the **2>** prompt enter:

go

 - A listing of connections to the database is displayed.
 - The listing includes the following types of data:
   - **Parameter Name** (i.e., number of user connections).
   - **Default.**
   - **Memory Used.**
   - **Config Value.**
   - **Run Value.**

9   At the **1>** prompt enter:

quit

 - The connection with the database is discontinued.

10   Compare the number of actual connections (results of **sp_who**) with the number of connections for which the database has been configured (results of **sp_configure "user connections"**).

11   If the number of actual connections is very close to the number of connections for which the database has been configured, notify the Database Administrator of the fact.

12   If the number of actual connections is **not** very close to the number of connections for which the database has been configured, compare the number of actual connections with the value for DBMaxConnections that the Database Administrator specified (Step 1).

13   If the number of actual connections is very close to the value for DBMaxConnections, notify the Database Administrator of the fact.

 - It may be advisable to increase the value assigned to the DBMaxConnections parameter in the Configuration Registry.

Table 13.7-9. Check Database Connections - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Identify the values for database parameters associated with the appropriate Production Planning application	**contact Database Administrator**
2	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
3	**isql –U <User ID> -S <Database Server>**	**enter text, press Enter**
4	**<database password>**	**enter text, press Enter**

Table 13.7-9. Check Database Connections - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
5	**sp_who**	**enter text, press Enter**
6	**go**	**enter text, press Enter**
7	**sp_configure "user connections"**	**enter text, press Enter**
8	**go**	**enter text, press Enter**
9	**quit**	**enter text, press Enter**
10	Compare the number of actual connections with the number of connections for which the database has been configured	**read text**
11	Notify the Database Administrator of the results	**contact Database Administrator**

13.7.2 Handle a Failure to Generate DPRs

There are several possible reasons for the Planning Subsystem to fail to generate DPRs. A failure to generate DPRs could be a normal consequence of one of the production rules. For example, if the Closest Granule production rule applies, there may be times when no acceptable input granules are found.   However, a failure to generate DPRs is most likely due to one of the following problems:

- Error creating a subscription for input data.
- **Production Request Editor** does not query DSS.
- Too many granules meet the criteria for input granules for a particular DPR.

A failure due to "too many granules" is caused when too many granules meet the criteria for input granules for a particular DPR.

- At PGE registration, the number of granules expected for each input ESDT is defined.
    - The definition includes the minimum number and the maximum number of granules expected.
- If the number of granules found is not between the minimum and maximum number, the request fails.
    - The failure occurs either when trying to save the production request or when the PCF file is generated (AutoSys pre-processing step).

The following procedures may be involved in investigating a failure to generate DPRs:

- Check the Production Request Editor ALOG File.
- Use ISQL to Check Database Tables.
- Check the PDPS Database for Causes of Failure to Generate DPRs.
- Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data.
- Check the Production Request Editor Debug File for Evidence of Metadata Queries.

13.7.2.1   Check the Production Request Editor ALOG File

Table 13.7-10 presents (in a condensed format) the steps required to check the **Production Request Editor** ALOG file. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the Planning/Management Workstation.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the planning log files (e.g., EcPlPREditor.ALOG).

3   At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlPREditor.ALOG).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4   Review the log file to determine whether the log contains an error message that indicates why the DPR generation failed.
- For example:

  Msg: PlPge::GetInputForDpr - Extr input to process DPR MoPGE01#2007081600OPS, for data type id MOD000#001, with logical id 599001. PlDataTypeReq has a scienceGroup of for this datatype. Expected 2 max inputs, but got 3. Priority : 2 Time : 07/09/99 17:10:52

  – In the example the **Production Request Editor** queried the PDPS database for granules that would satisfy the data needs for the DPR and found three granules instead of the two it expected (i.e., "Expected 2 max inputs, but got 3").
  – Consequently, the Production Request failed due to "too many granules."
- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

5   If the log does **not** contain an error message that indicates why the DPR generation failed, perform the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).

6   If there is an error in the log file indicating that too many granules meet the criteria for input granules for a particular DPR, go to the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).

7   If there is an error in the log file indicating that an entry could not be found for a specific ESDT in the appropriate database, notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.
- If no entry can be found for the ESDT in question it indicates either of the following conditions:
  – The ESDT was not installed.
  – The ESDT was not installed properly.

8   If there is an error in the log file indicating a subscription error, notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.
- If subscription errors exist in the log, it is most likely that the ESDT in question (for the input) was either not installed or improperly installed.
- Notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.

9   If there is an error in the log file indicating communication problems, notify the Operations Controller/System Administrator of the suspected communication problems.

10   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

11   If there is an error in the log file indicating database problems, at the command line prompt (for the Queuing Server) enter:

ps –ef | grep sybase
- A listing of processes containing the string "sybase" is displayed:
- If the PDPS database server is running, the following type of message would be displayed:

  /usr/ecs/OPS/COTS/sybase/bin/dataserver -d/dev/rdsk/c0t2d0s4 -sx0sps02_srvr -e/
- If the PDPS database server were **not** running, the following type of message only would be displayed:

  cmshared 15516 15405 0 14:20:33 pts/6   0:00 grep sybase

12   If the database servers are **not** running, notify the Database Administrator of the problem.

13   If there is an error in the log file indicating database problems and the database servers are running, go to the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).
  - Check for lack of (or corruption of) data in the database.

14   If there is an error in the log file indicating lack of disk space, access a terminal window logged in to the Planning/Management Workstation.
  - Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
  - For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

15   If there is an error in the log file indicating lack of disk space, at the command line prompt (for the Planning/Management Workstation) enter:

  cd /<path>

  - **<path>** indicates the directory path to a directory (on the disk that is full) from which files can be deleted.

16   If there is an error in the log file indicating lack of disk space, at the command line prompt (for the Planning/Management Workstation) enter:

  rm <file name 1> … <file name x>

  - **<file name 1> … <file name x>** represent unnecessary files to be removed.
  - Use wild cards if appropriate.
    - For example:

      rm *.hdf

      would prompt the system to remove all files with a suffix of .hdf.

17   If removing files from a full disk, respond appropriately to system prompts.
  - For example:

  rm: remove trashfile.rs (yes/no)? y

18   If there is a recurring disk space problem, notify the Operations Controller/System Administrator of the problem.

19   If the log contains an error message which indicates that a configuration item in another subsystem may be the source of the problems, consult with the relevant technician (e.g., Distribution Technician) to request assistance in isolating the problem.

20 When the problem has been corrected, use the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3) to re-create the Production Request that led to discovery of the problem.

Table 13.7-10. Check the Production Request Editor ALOG File - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Review the log file	**read text**
5	Check the PDPS database for causes of failure to generate DPRs (if applicable)	Use procedure in Section 13.7.2.3
6	Notify the Science Data Specialist to verify that the corresponding ESDT(s) has/have been properly installed (If applicable)	**contact Science Data Specialist**
7	Notify the Operations Controller/System Administrator of suspected communication problems (if applicable)	**contact Operations Controller**
8	UNIX window (Queuing Server host)	**single-click** or use procedure in Section 13.2.1
9	**ps –ef	grep sybase**	**enter text, press Enter**
10	Notify the Database Administrator if the database servers are not running	**contact Database Administrator**
11	Check the PDPS database for causes of failure to generate DPRs (if applicable)	Use procedure in Section 13.7.2.3
12	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
13	**cd /<path>** [for cleaning up disk]	**enter text, press Enter**
14	**rm <file name 1> … <file name x>**	**enter text, press Enter**
15	Consult with other technicians to isolate the problem (as needed)	**contact technician(s)**
16	Create a new Production Request when the problem has been corrected	Use procedure in Section 13.2.3

13.7.2.2　　Use ISQL to Check Database Tables

The PDPS database is the repository of data concerning PGEs, Production Requests, Data Processing Requests, Production Strategies, Production Plans and other production-related data. The Subscription Server (SUBSRV) database contains data concerning subscriptions.

The data stored in the database can be checked using either a database browser or isql commands. The procedure in this section describes how to check the tables using isql commands.

Table 13.7-11 presents (in a condensed format) the steps required to use isql to check database tables. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the appropriate host.
        • Examples of Queuing Server host names include **e0sps04 and l0sps03**.
        • Examples of Subscription Server host (Sun Consolidation Internal Server host) names include **e0acs11 and l0acs03**.
        • For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

        isql –U <user ID> -S <database server>

        • **<user ID>** is the database user's identification; e.g., **pdps_role**.
        • **<database server>** is the database server; e.g., **x0sps02_srvr**.
        • For example:

        isql –U pdps_role –S x0sps02_srvr

3   At the **Password:** prompt enter:

        <database password>

        • **<database password>** is the password for logging in to the database using the specified **<user ID>**.
        • A **1>** prompt is displayed, indicating that a connection has been made with the database.

4   At the **1>** prompt enter:

        use <database name>

        • The **<database name>** is likely to be one of the following names:
            – **pdps** [OPS mode].
            – **pdps_TS1** [TS1 mode].
            – **pdps_TS2** [TS2 mode].

5   At the **2>** prompt enter:

        go

6   At the **1>** prompt enter:

select * from <table name>

- Alternatively, enter:

    select <column name> from <table name>

- Another alternative:

    select <column name1>,<column name2>[,<column name3>,…] from <table name>

7   At the **2>** prompt enter:

go

- Table contents are displayed.
    - If * was specified, all entries in the table are displayed.
    - If specific column names were entered, the data associated with those columns only are displayed.

8   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

Table 13.7-11. Use ISQL to Check Database Tables - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (appropriate host)	**single-click** or use procedure in Section 13.2.1
2	**isql –U <user ID> -S <database server>**	**enter text, press Enter**
3	**<database password>**	**enter text, press Enter**
4	**use <database name>**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**select * from <table name>**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	**quit**	**enter text, press Enter**

13.7.2.3   Check the PDPS Database for Causes of Failure to Generate DPRs

The PDPS database is a useful resource for troubleshooting a failure to generate DPRs. Certain values must be entered in tables in the PDPS database during the DPR generation process in order for the DPRs to be successfully generated.

- The subscriptionFlag values (PlDataTypeMaster table) for all the data types (ESDTs) needed for the PGE are set.

- The Science Data Server UR is entered in the dataServUrString column (PlDataTypeMaster table) for all data types (ESDTs) needed for the PGE.
- If the PGE specified in the Production Request requires static files, the URs for the static files are included in PlDataGranuleShort table (universalReference column).

Table 13.7-12 presents (in a condensed format) the steps required to check the PDPS database for causes of failure to generate DPRs. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Log in to the PDPS database.
- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2   At the **1>** prompt enter:

select dataTypeId,subscriptionFlag from PlDataTypeMaster

- Prepare a request to view PlDataTypeMaster table subscriptionFlag column values for all data types (dataTypeId column).

3   At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataTypeMaster** table are displayed.
- During DPR generation, the DPR executable should turn the subscriptionFlag for all the data types (ESDTs) needed for the PGE from zero to non-zero.
- If the subscriptionFlag values for the data types did not turn to non-zero, subscription trouble is indicated.

4   If the subscriptionFlag value for any ESDT needed for the PGE is **zero**, make a note of the fact for subsequent reporting of the problem.

5   At the **1>** prompt enter:

select dataTypeId,dataServUrString from PlDataTypeMaster

- Prepare a request to view PlDataTypeMaster table dataServUrString column values for all data types.

6   At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataTypeMaster** table are displayed.
- During DPR generation, the DPR executable should turn the dataServUrString for all the ESDTs needed for the PGE from "NULL" to the UR value for Science Data Server (e.g., UR:15:DsShSciServerUR:13:[MDC:DSSDSRV]).

7   If the dataServUrString value for any ESDTs needed for the PGE does not have the UR value for the Science Data server (e.g., is NULL), make a note of the fact for subsequent reporting of the problem.

8   If the PGE specified in the Production Request requires static files, at the **1>** prompt enter:

select dataTypeId,universalReference from PlDataGranuleShort

- Prepare a request to view PlDataGranuleShort table universalReference column for the URs of the necessary static files.
- If there is data on many data types in the PlDataGranuleShort table, it may be advisable to limit the search, for example:

1> select dataTypeId,universalReference from PlDataGranuleShort where dataTypeId = "AM1ATTNF#001"

  - In the example **AM1ATTNF#001** is the data type being checked.

9   At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataGranuleShort** table are displayed.
- If the PGE requires static files, the URs for the static files must be included in the PlDataGranuleShort table in order for a DPR to be successfully generated.
- For dynamic granules, the corresponding UR values may become available during DPR generation; however, if the dynamic granules' URs do not become available during DPR generation, there is no effect on DPR generation.

10   If the PGE requires static files, and the URs for the static files are **not** in the PlDataGranuleShort table, make a note of the fact for subsequent reporting of the problem.

11   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

12   If any problems were noted in the PDPS database, report the problem(s) to the SSI&T team and/or call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
- There may be a problem with PGE registration (especially if no other PGEs are affected) or other problem associated with the SSI&T process.

Table 13.7-12. Check the PDPS Database for Causes of Failure to Generate DPRs - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**select dataTypeId,subscriptionFlag from PlDataTypeMaster**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select dataTypeId,dataServUrString from PlDataTypeMaster**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**select dataTypeId,universalReference from PlDataGranuleShort**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	**quit**	**enter text, press Enter**
9	Report all problems to the SSI&T team and/or call the help desk and submit a trouble ticket	**contact SSI&T team**

13.7.2.4   Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data

If the Production Request Editor reports a failure creating a DPR that is subject to the Metadata Query production rule, the process for responding to the problem consists of the following activities:

- Check the "debug" log for evidence that the Science Data Server was queried for metadata values.
- If there is no message in the debug log indicating that the Science Data Server was queried for metadata values, the SSI&T team should check for an error in the appropriate PGE science metadata ODL file.

13.7.2.5   Check the Production Request Editor Debug File for Evidence of Metadata Queries

Table 13.7-13 presents (in a condensed format) the steps required to check the Production Request Editor debug file for evidence of metadata queries. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the Planning/Management Workstation.
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the planning log files (e.g., EcPlPREditorDebug.log).

3   At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlPREditorDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4   Review the log file to determine whether the log contains a message that indicates why the DPR generation failed.

- When the Production Request Editor queries the Science Data Server to search for the PGE's input(s), the query should include the metadata value(s) desired by the PGE.
- Note that events are recorded in the debug log in chronological order and are preceded by a time and date stamp.
- If there is no message which indicates that the Science Data Server was queried for the metadata value(s) desired by the PGE, there might be an error in the PGE Metadata ODL File for the PGE.
- To exit from **pg** at the **:** prompt enter:

  q

  − The command line prompt is displayed.

5   If there was no message in the debug log indicating that the Science Data Server was queried for metadata values, submit a request to the SSI&T team to check for an error in the appropriate PGE science metadata ODL file and re-register the PGE.

- After the SSI&T team has edited the ODL file and re-registered the PGE it should be possible to successfully create DPRs for the PGE.

Table 13.7-13. Check the Production Request Editor Debug File for Evidence of Metadata Queries - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**

Step	What to Enter or Select	Action to Take
4	Review the log file	**read text**
5	Ask the SSI&T team to check for errors in the PGE science metadata ODL file (if there was no metadata query message in the debug log)	**contact SSI&T team**

13.7.3 Respond to PR or DPR Deletion that Hangs

When deleting DPRs or PRs from the Production Request Editor, the Job Management Server and Deletion Server (both in DPS) are called to clean up all PDPS database tables associated with the DPR or PR. It is possible that any failure or "hung" condition can be attributed to the Production Request Editor itself or the Deletion or Job Management Server. If the servers are functioning properly, there are two other possible causes:

- The database is locked.
- There may be resource locks in the PDPS database.

Accordingly, the following procedures may be involved in responding to PR or DPR deletion that hangs:

- Check for Database Deadlocks.
- Check for Resource Locks in the PDPS Database.

13.7.3.1   Check for Database Deadlocks

A deadlock occurs when a database transaction locks a record that another transaction needs and the second transaction locks the record that first transaction needs. Each program must wait until the other completes. However, neither can complete (because each is waiting for the other) so both end up waiting indefinitely.

Table 13.7-14 presents (in a condensed format) the steps required to check for database deadlocks. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Log in to the PDPS database.
- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2   At the **1>** prompt enter:

sp_lock

3   At the **2>** prompt enter:

go

- Results displayed include the following features:
    - **spid** column shows the process id. The database user that owns a process can be determined using the **sp_who** isql command.
    - **locktype** column indicates a problem if the entry starts with "Ex_" (exclusive).
    - **table_id** column identifies the table that the corresponding spid has locked. The name of the table can be determined using the **select** command [i.e., **select object_name (table_id)**].

4   At the **1>** prompt enter:

select object_name (<table id>)

- For example, to check the exclusive locks related to spid 24, table ID 197575742, enter:

    1> select object_name (197575742)

5   At the **2>** prompt enter:

go

- The object name is displayed (e.g., PlDprData).

6   At the **1>** prompt enter:

sp_who

7   At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns:
    - **spid.**
    - **status.**
    - **loginame.**
    - **hostname.**
    - **blk.**
    - **dbname.**
    - **cmd.**

8   Analyze the results of the request.
- The **blk** column shows the spid of the process that is doing the blocking.
- The **cmd** column shows the command that the blocked process is trying to complete.

9   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

10   If there is a deadlock in the database, ask the Operations Controller to bounce the server
that is causing the deadlock.

11   If there is no deadlock, perform the **Check for Resource Locks in the PDPS Database**
procedure (Section 13.7.3.1).

Table 13.7-14. Check for Database Deadlocks - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**sp_lock**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select object_name (<table id>)**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**sp_who**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	Analyze the results of the request	**read text**
9	**quit**	**enter text, press Enter**
10	Ask the Operations Controller to bounce any server that is causing a deadlock (if applicable)	**contact Operations Controller**
11	Check for resource locks in the PDPS database (if there is no deadlock)	Use procedure in Section 13.7.3.1

13.7.3.1   Check for Resource Locks in the PDPS Database

Resource locks used to occur if there was an attempt to delete DPRs/PRs while their
corresponding jobs were still running in AutoSys or jobs had been explicitly killed before the
DPRs/PRs were deleted.   However, resource locking has been removed for all Resource
Management calls (e.g., for allocating CPUs and disk space). The locks have been replaced with
the following features:

- Sybase stored procedures that use transactions.
- Database triggers.

Resource locking is still used for disk space reclamation. Momentary system interruptions occur
during the process of disk space reclamation. The interruptions may happen several times a day.
The system may look like it is "hung" during such periods. The procedure that follows should be
performed to verify that disk space reclamation is proceeding normally:

Although the procedure for checking for resource locks in the PDPS database includes the use of isql commands, an acceptable alternative is to use a database browser to check the contents of the DpPrResourceLock table.

Table 13.7-15 presents (in a condensed format) the steps required to check for resource locks in the PDPS database. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.
- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

select * from DpPrResourceLock

- Prepare a request to view the contents of the DpPrResourceLock table.

3 At the **2>** prompt enter:

go

- The contents of the **DpPrResourceLock** table are displayed.
- The listing includes data in the following columns:
  - **jobId.**
  - **priority.**
  - **ecsUnit.**
  - **attempts.**
  - **state.**
  - **pid.**
  - **queuePosition.**

4 Analyze the results of the request.
- A jobId with a state $<> 0$ would indicate a resource lock.
  - If there are entries in the **DpPrResourceLock** table and there are no other jobs running in AutoSys, all entries in the table need to be deleted before the DPR/PR deletion can complete.
  - If other jobs (DPRs) are currently being executed in AutoSys and the other jobs should not be deleted, the entries in the table that need to be deleted are those related to the job to be deleted only. The entries concerning the other (running) jobs must be left in the table.
- If there is no evidence of a resource lock, go to Step 8.

5   If all entries in the DpPrResourceLock table are to be deleted, at the **1>** prompt enter:

delete DpPrResourceLock

- Go to Step 7.

6   If some (but not all) entries in the DpPrResourceLock table are to be deleted, at the **1>** prompt enter:

delete DpPrResourceLock where jobId like "<job Id>"

- **<job Id>** specifies the job whose entries are to be deleted.

7   At the **2>** prompt enter:

go

- Entries in the **DpPrResourceLock** table are deleted.
- The DPR/PR deletion that was delayed by the resource lock should go to completion.

8   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.
- If entries were deleted from the DpPrResourceLock table the procedure is finished; otherwise, continue with Step 9.

9   Access a terminal window logged in to the Queuing Server.

- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

10   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the data processing log files (e.g., EcDpPrJobMgmt.ALOG, EcDpPrDeletion.ALOG).

11   At the command line prompt enter:

tail -f <job Id>.err

- **<job Id>.err** refers to the data processing log file to be reviewed.

12   Observe the log file to determine whether entries are being made in the file.

- If messages are being entered in the log file, there is probably no resource lock.

13   To quit tailing the log in the terminal window enter:

Ctrl-C

- A command line prompt is displayed in the terminal window.

14   Ensure that it is possible to connect to the necessary hosts and servers.
- For detailed instructions refer to the **Check Connections To Hosts/Servers** procedure (Section 13.7.1.1).

15   If no there is no database deadlock or resource lock and the Data Processing Subsystem servers (especially Deletion Server and Job Management Server) are up, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-15. Check for Resource Locks in the PDPS Database - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**Select * from DpPrResourceLock**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	Analyze the results of the request.	**read text**
5	**delete DpPrResourceLock** or **delete DpPrResourceLock where jobId like "<job Id>"** (as applicable)	**enter text, press Enter**
6	**go** (if applicable)	**enter text, press Enter**
7	**quit**	**enter text, press Enter**
8	UNIX window (Queuing Server)	**single-click** or use procedure in Section 13.2.1
9	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
10	**tail -f <job Id>.err**	**enter text, press Enter**
11	Observe the log file (Are entries are being made in the log file?)	**read text**
12	**Ctrl-C**	**enter text**
13	Check connections to hosts/servers	Use procedure in Section 13.7.1.1
14	Call the help desk and submit a trouble ticket (if applicable)	Use procedure in Chapter 8

13.7.4 Respond to DPR Deletion that Fails

When deleting DPRs or PRs from the Production Request Editor, both the Job Management Server and Deletion Server (both DPS servers) are called to clean up all PDPS database tables associated with the DPR or PR. It is possible that any failure or "hung" condition can be attributed to the Production Request Editor itself or to the Job Management Server or Deletion Server.

Table 13.7-16 presents (in a condensed format) the steps required to respond to DPR deletion that fails. If you are already familiar with the procedures, you may prefer to use the quick-step

table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04 and l0sps03**.
- For detailed instructions refer to the **Log in to System Hosts** procedure (Section 13.2.1).

2   At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the Data Processing Subsystem log files (e.g., EcDpPrDeletionDebug.log).

3   If there is a Deletion Server Debug log, at the command line prompt enter:

pg <file name>

- **<file name>** refers to the Data Processing Subsystem log file to be reviewed (e.g., EcDpPrDeletionDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4   If there is a Deletion Server Debug log, review the log file for error messages.
- For example:

   **Could not make database interface to DpPrFile
   Aborting: will report error to client.**

- The message in the example indicates a compilation problem in Deletion Server so Deletion Server cannot communicate with the PDPS database.
  - When Deletion Server was compiled the RogueWave database (RWDB) libraries were not linked correctly.
  - The most likely cause is the fact that the Deletion Server Imakefile was changed so that the DPS make.options was included.
- To exit from **pg** at the **:** prompt enter:

   q

  - The command line prompt is displayed.

5   If an error message of the type shown in the previous step was present in the log, notify the Operations Controller/System Administrator of the problem.

Table 13.7-16. Respond to DPR Deletion that Fails - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Review the log file	**read text**
5	Notify the Operations Controller/System Administrator of the problem (if applicable)	**contact Operations Controller**

13.7.5 Handle a DPR Scheduling Failure

There are three principal sources of information concerning failures to activate DPRs from the Planning Workbench:

- The Planning Workbench GUI itself may provide an error message.
    − The message would indicate either success or failure.
    − It would not indicate why the request failed.
- The Planning Workbench Message Handler (the window that comes up before the Planning Workbench and to which the Workbench logs various messages).
    − Any error encountered should be logged to the Message Handler.
- The Planning Workbench logs (debug log and ALOG).
    − The logs are located in the /usr/ecs/<MODE>/CUSTOM/logs directory and begin with "EcPlWb" (e.g., EcPlWbDebug.log, EcPlWb.ALOG).
    − The logs can further refine the cause of the error.

If job activation fails from the Planning Workbench, one of the following errors is likely to be reported:

- "DPR Validation Failed"
- "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}"

Accordingly, the following procedures may be involved in handling a DPR scheduling failure:

- Respond to a "DPR Validation Failed" Error.
- Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error.

13.7.5.1   Respond to a "DPR Validation Failed" Error

There are two conditions that may cause a "DPR Validation Failed" error to be reported:

- Performance data for the PGE are missing from the PDPS PlPgePerformance database table.
- Resource information for the PGE is missing from the PDPS PlResourceRequirement database table.

Consequently, it is necessary to check the appropriate PDPS database tables to determine whether the necessary PGE information is in the database.

Although the procedure for responding to a "DPR Validation Failed" error includes the use of isql commands, an acceptable alternative is to use a database browser to check the contents of the PlPerformance and PlPgePerformance tables.

Table 13.7-17 presents (in a condensed format) the steps required to respond to a "DPR Validation Failed" error. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Log in to the PDPS database.
        - Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2   At the **1>** prompt enter:

        select * from PlPgePerformance

        - Prepare a request to view the contents of the PlPgePerformance table.

3   At the **2>** prompt enter:

        go

        - The contents of the **PlPgePerformance** table are displayed.
        - The listing includes data in the following columns:
            - **pgeId.**
            - **cpuTime.**
            - **pgeElapsedTime.**
            - **dprElapsedTime.**
            - **maxMemory.**
            - **faults.**
            - **swaps.**
            - **blockInputOperation.**
            - **blockOutputOperation.**
            - **runCpuTime.**
            - **runMaxMemory.**
            - **runPgeElapsed.**
            - **runDprElapsed.**
            - **runFaults.**
            - **runSwaps.**
            - **runBlockInOperation.**
            - **runBlockOutOperation.**
            - **sharedMemory.**
            - **runSharedMemory.**

- For the PGE(s) that is (are) not schedulable, verify there are non-zero values for the entries in the table.
  - Zero may be appropriate for some columns but not for all.

4   At the **1>** prompt enter:

select * from PlResourceRequirement

- Prepare a request to view the contents of the PlResourceRequirement table.

5   At the **2>** prompt enter:

go

- The contents of the **PlResourceRequirement** table are displayed.
- The listing includes data in the following columns:
  - **sswId.**
  - **string.**
  - **numOfCPUs.**
  - **computer.**
  - **diskSpace.**
  - **topLevelShellName.**
  - **exeTarFileDiskSpace.**
  - **mcfName.**
  - **ramSize.**
  - **exeUntarFileDiskSpace.**
  - **exeTarUR.**
  - **pgeId.**
  - **toolkitArchitecture.**
  - **pgeCommands.**
- For the PGE(s) that is (are) not schedulable, verify that there are non-zero values for the entries in the table.
  - Zero may be appropriate for some columns but not for all.

6   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

7   If the entries in either the **PlPgePerformance** or **PlResourceRequirement** table are not appropriate, make a request to the SSI&T team to have the correct values entered.

8   Delete the affected DPR(s) as described in the **Delete a Data Processing Request** procedure (Section 13.2.8).

9   When the SSIT team has completed updating the PGE information in the PDPS database, create a new Production Request to replace the affected DPR(s) as described in the

Create a New Production Request Using the Production Request Editor GUI
procedure (Section 13.2.3).

10   Create and activate a new Production Plan (to send the replacement DPR(s) to data
        processing) as described in the **Create a New Production Plan** procedure
        (Section 13.4.2).

Table 13.7-17. Respond to a "DPR Validation Failed" Error -
Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**select * from PlPgePerformance**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select * from PlResourceRequirement**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**quit**	**enter text, press Enter**
7	Ask the SSI&T team to have the correct values entered in the database tables (if applicable)	**contact SSI&T team**
8	Delete the affected DPR(s)	Use procedure in Section 13.2.8
9	Create a new Production Request to replace the affected DPR(s) when the SSIT team has completed updating the PGE information in the PDPS database	Use procedure in Section 13.2.3
10	Create and activate a new Production Plan (to send the replacement DPR(s) to data processing)	Use procedure in Section 13.4.2

13.7.5.2   Respond to an "Information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error

When an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" error is reported on the Message Handler, one of two problems is likely:

- A predecessor DPR exited in a "bad" state.
    - The unschedulable DPR depends on the predecessor DPR for input data.
    - The predecessor DPR is in either a completed or an error status, or was not included in the activated plan.
- The current DPR has a "QA_FAILURE."
    - A metadata check or query for the DPR failed so the DPR cannot be scheduled.

Consequently, it is necessary to check the appropriate PDPS database tables and possibly the Planning Workbench debug log to localize the problem.

Although the procedure for responding to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" error includes the use of isql

commands, an acceptable alternative is to use a database browser to check the contents of the appropriate tables.

Table 13.7-18 presents (in a condensed format) the steps required to respond to a "DPR Validation Failed" error. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1   Log in to the PDPS database.
   - Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2   At the **1>** prompt enter:

   select * from PlDataProcessingRequest

   - Prepare a request to view the contents of the PlDataProcessingRequest table.
   - If there is data on many DPRs in the PlDataProcessingRequest table, it may be advisable to limit the search, for example:

   1> select * from PlDataProcessingRequest where dprId like "AM1Eph#30002282200TS2"

   - In the example **AM1Eph#30002282200TS2** is the predecessor DPR being checked.

3   At the **2>** prompt enter:

   go

   - The contents of the **PlDataProcessingRequest** table are displayed.
   - The listing includes data in the following columns:
     - **dprId.**
     - **productionRequestId.**
     - **priority.**
     - **predictedStart.**
     - **actualStart.**
     - **completionState.**
     - **pgeId.**
     - **baselineTime.**
     - **name.**
     - **dataStartTime.**
     - **dataStopTime.**
     - **sswId.**
     - **dprCollectionId.**
     - **alarmTime.**
     - **lateMessagesSent.**
     - **autosysId.**

- **tileId.**
- **timeStamp.**
- For the predecessor DPR to the unschedulable DPR, verify that the **completionState** is **SUCCESS**.

4   To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

5   If the **completionState** for the predecessor DPR to the unschedulable DPR is **not SUCCESS**, perform Steps 6 through 8; otherwise, go to Step 9.
- For example, if **completionState** is **FAILEDPGE** or **CANCELED**, perform Steps 6 through 8.

6   Delete the unschedulable DPR(s) and its predecessor(s) as described in the **Delete a Data Processing Request** procedure (Section 13.2.8).

7   Create new Production Request(s) to replace the affected DPR(s) as described in the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).

8   Create and activate a new Production Plan (to send the replacement DPR(s) to data processing) as described in the **Create a New Production Plan** procedure (Section 13.4.2).
- End of procedure.

9   If the **completionState** for the predecessor DPR to the unschedulable DPR is **SUCCESS**, at the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- For**<MODE>** type the desired operating mode.
- Change directory to the directory containing the appropriate log files (e.g., EcPlWbDebug.log).

10   At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlWbDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

11 Review the log file to determine whether the log contains an error message (concerning the unschedulable DPR) that indicates why the DPR generation failed.

- Look for an entry for the unschedulable DPR that contains either "QA_FAILURE" or "INVALID."
  - "QA_FAILURE" indicates a problem with a metadata check/query so the DPR cannot be scheduled.
  - "INVALID" indicates that DPR validation failed.
- To exit from **pg** at the **:** prompt enter:

  q
  - The command line prompt is displayed.

12 If the log contains an error message that includes the term "INVALID", perform the **Respond to a "DPR Validation Failed" Error** procedure (Section 13.7.5.1).

13 If the log contains a "QA_FAILURE" error message, log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

14 At the **1>** prompt enter:

select * from PlDprData

- Prepare a request to view the contents of the PlDprData table.
- If there is data on many DPRs in the PlDprData table, it may be advisable to limit the search, for example:

  1> select * from PlDprData where dprId like "ACT#syn1#004130123TS2"

  - In the example **ACT#syn1#004130123TS2** is the predecessor DPR being checked.

15 At the **2>** prompt enter:

go

- The contents of the **PlDprData** table are displayed.
- The listing includes data in the following columns:
  - **dprId.**
  - **granuleId.**
  - **logicalId.**
  - **primaryType.**
  - **accepted.**
  - **theOrder.**
  - **type.**
  - **temporalFlag.**
  - **timeWait.**
  - **ioFlag.**

- − **timerExp.**
- − **timerStart.**
- − **numNeeded.**
- − **waitForFlag.**
- − **linkId.**
- − **minGranReq.**
- Note that for each DPR there are separate entries for the different data types.
- For all entries for the unschedulable DPR, verify that the values in the **accepted** column are **not** equal to 2.
  - − A value of 2 in the **accepted** column indicates that the metadata check/query for the corresponding input (**granuleId** column) failed.

16   At the **1>** prompt enter:

select * from PlEsdtParam

- Prepare a request to view the contents of the PlEsdtParam table.
- If there is data on many DPRs in the PlEsdtParam table, it may be advisable to limit the search, for example:

   1> select * from PlEsdtParam where dataTypeId like "AST_L1B#001"

   - − In the example **AST_L1B#001** is the data type being checked.

17   At the **2>** prompt enter:

go

- The contents of the **PlEsdtParam** table are displayed.
- The listing includes data in the following columns:
  - − **dataTypeId.**
  - − **paramName.**
  - − **paramType.**
  - − **containerName.**
  - − **secondParm.**
  - − **secondValue.**
- Verify that there are entries for the appropriate **dataTypeId** in the PlEsdtParam table.

18   If no entry for the unschedulable DPR has a value equal to 2 in the **accepted** column, at the **1>** prompt enter:

select * from PlEsdtParmValues

- Prepare a request to view the contents of the PlEsdtParmValues table.

19   At the **2>** prompt enter:

go

- The contents of the **PlEsdtParmValues** table are displayed.

- The listing includes data in the following columns:
  - **esdtParmName.**
  - **esdtParmVal.**
  - **granuleId.**
  - **secondParm.**
  - **secondValue.**
- Verify that there are entries for the appropriate data type (in the **granuleId** column) in the PlEsdtParmValues table.

20   If there are entries for the appropriate data type in the **PlEsdtParam** and **PlEsdtParmValues** tables, at the **1>** prompt enter:

select * from PlMetadataChecks

- Prepare a request to view the contents of the PlMetadataChecks table.
- If there is data on many data types in the PlMetadataChecks table, it may be advisable to limit the search, for example:

  1> select * from PlMetadataChecks where dataTypeId like "AST_L1B#001"

  - In the example **AST_L1B#001** is the data type being checked.

21   At the **2>** prompt enter:

go

- The contents of the **PlMetadataChecks** table are displayed.
- The listing includes data in the following columns:
  - **pgeId.**
  - **dataTypeId.**
  - **paraName.**
  - **logicalId.**
  - **type.**
  - **operator.**
  - **value.**
  - **ioFlag.**
  - **queryFlag.**
  - **queryType.**
  - **secondParm.**
  - **secondValue.**
- Verify that the entries for the appropriate data type in the PlEsdtParam and PlEsdtParmValues tables match the expected values in the PlMetadataChecks table.

22   If an entry for the unschedulable DPR has a value equal to 2 in the **accepted** column of the **PlDprData** table (Step 15), make a record of the fact that the PGE cannot be run on the specified input data.

- The metadata check/query for the corresponding input failed.

- It may be possible to either reingest or reprocess the input data that failed the metadata check or query.
  - Steps 17 through 21 can give indications as to which applies, based on the data type(s) involved.

23 If no problem is identified in the preceding steps, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-18. Respond to an "Information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**Select * from PlDataProcessingRequest**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**quit**	**enter text, press Enter**
5	Go to Step 9 if the **completionState** for the predecessor DPR to the unschedulable DPR is "SUCCESS"	
6	Delete the unschedulable DPR(s) and its predecessor(s)	Use procedure in Section 13.2.8
7	Create new Production Request(s) to replace the affected DPR(s)	Use procedure in Section 13.2.3
8	Create and activate a new Production Plan [end of procedure]	Use procedure in Section 13.4.2
9	**cd /usr/ecs/<MODE>/CUSTOM/logs** (if applicable)	**enter text, press Enter**
10	**pg <file name>**	**enter text, press Enter**
11	Review the log file	**read text**
12	Respond to the "DPR Validation Failed" error (if applicable)	Use procedure in Section 13.7.5.1
13	Log in to the PDPS database (if applicable)	Use procedure in Section 13.7.2.2
14	**select * from PlDprData**	**enter text, press Enter**
15	**go**	**enter text, press Enter**
16	**select * from PlEsdtParam**	**enter text, press Enter**
17	**go**	**enter text, press Enter**
18	**select * from PlEsdtParmValues** (if applicable)	**enter text, press Enter**
19	**go**	**enter text, press Enter**
20	**select * from PlMetadataChecks** (if applicable)	**enter text, press Enter**
21	**go**	**enter text, press Enter**
22	Make a record of the fact that the PGE cannot be run on the specified input data (if applicable)	**enter text**

Table 13.7-18. Respond to an "Information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
23	Call the help desk and submit a trouble ticket in accordance with site Problem Management policy (if applicable)	Use procedure in Chapter 8

This page intentionally left blank.