Contents

11. SSI&T Operational Procedures

11.1 Preparation....11-2

11.1.1 Creating a New Earth Science Data Type (ESDT)....11-2

11.1.2 Validating Granule Level Metadata....11-2

11.1.3 Installing an Earth Science Data Type (ESDT) on the Data Server....11-2

11.1.4 Validating Successful ESDT Installation....11-2

11.1.5 Creating Metadata Configuration Files (MCF'....11-2

11.2 Acquiring the Delivered Algorithm Package (DAP)....11-2

11.2.1 Registering a Subscription for a DAP....11-2

11.2.2 Notification of DAP Arrival....11-2

11.2.3 Acquiring a DAP from Data Server....11-2

11.2.4 Acquiring a DAP via FTP....11-2

11.3 Configuration Management....11-2

11.3.1 Creating and Using a View in ClearCase....11-2

11.3.2 Importing Multiple Files into ClearCase from a Directory Structure....11-4

11.3.3 Entering a Single File into ClearCase....11-5

11.3.4 Entering a New Directory into ClearCase....11-6

11.3.5 Checking Out an Element from ClearCase....11-7

11.3.6 Checking a Revised Element into ClearCase....11-7

11.3.7 Applying Labels to ClearCase Elements....11-8

11.4 SSIT Manager GUI....11-8

11.4.1 General Setup of the SSIT Manager....11-8

11.4.2 Setup of Checklist for SSIT Manager....11-11

11.4.3 Running the SSIT Manager for the First Time....11-12

11.4.4 Routine Running of the SSIT Manager....11-14

11.5 Standards Checking....11-15

11.5.1 Checking ESDIS Standards Compliance: FORTRAN 77....11-15

11.5.2 Checking ESDIS Standards Compliance: Fortran 90....11-18

11.5.3 Checking ESDIS Standards Compliance: C....11-18

11.5.4 Checking ESDIS Standards Compliance: Ada....11-20
11.5.5 Prohibited Function Checker....11-21
11.5.6 Checking Process Control Files....11-22
11.5.7 Extracting Prologs....11-23
11.6 Compiling and Linking....11-23
11.6.1 Updating the Process Control File (PCF)....11-23
11.6.2 Compiling Status Message Facility (SMF) Files....11-27
11.6.3 Setting up a SDP Toolkit Environment....11-30
11.6.4 Compiling a PGE and Linking With SCF Version of SDP Toolkit....11-33
11.6.5 Compiling a PGE and Linking with DAAC Version of SDP Toolkit....11-36
11.7 Running a PGE in a Simulated SCF Environment at the DAAC....11-39
11.7.1 Setting Up the Environment for Running the PGE....11-39
11.7.2 Running and Profiling the PGE....11-39
11.8 Updating the PDPS Database and Data Server....11-39
11.8.1 Updating the Science Metadata in the PDPS Database....11-39
11.8.2 Updating the ESDT Metadata in the PDPS Database....11-39
11.8.3 Updating the Operational Metadata in the PDPS Database....11-39
11.8.4 Inserting Test Data Files Into the Data Server....11-39
11.8.5 Inserting Static Files into the Data Server....11-39
11.8.6 Inserting the Science Software Executable Package (SSEP) into the Data Server....11-39
11.9 PGE Planning and Processing....11-39
11.9.1 Using the Planning Workbench....11-39
11.9.2 Using the Production Request Editor....11-39
11.9.3 Registering a Subscription for Test Output Files....11-39
11.9.4 Monitoring Production....11-39
11.9.5 Acquiring the Test Output Files from the Data Server....11-39
11.10 File Comparison....11-39
11.10.1 Using the HDF File Comparison Tool GUI....11-39
11.10.2 Using theHDF File Comparison Tool hdiff....11-40
11.10.3 Using the ASCII File Comparison Tool....11-40
11.10.4 Using the Binary File Comparison Tool....11-41

11.11 Data Visualization 11-42
11.11.1 Viewing Product-Created Metadata Using the EOSView Tool 11-42
11.11.2 Viewing Product Data with the EOSView Tool 11-44
11.11.3 Viewing Product Data with the IDL Tools 11-44
11.12 Post-Testing Activities 11-44
11.12.1 Validating Inventory/Granule Metadata Updates to the Data Server Database 11-44
11.12.2 promoting the PGE from Test to Production 11-44
11.12.3 Reporting Science Software Problems 11-44
11.12.4 Reporting ECS Software Problems 11-44
11.13 Science Software Archive Package (SSAP) 11-44
11.13.1 Assembling Components of the SSAP 11-44
11.13.2 Creating and Inserting the SSAP into the Data Server 11-44
11.14 Troubleshooting and General Investigation 11-44
11.14.1 Examining PGE-Produced Log Files 11-44
11.14.2 Examining the MSS Log File 11-48
11.14.3 Using PDPS Prototype-Related Scripts and Examining Message Files 11-49
11.15 Miscellaneous 11-54
11.15.1 Setting Up the Release A Bulletin Board Service 11-54
11.15.2 Obtaining Information from the Bulletin Board Service 11-54
11.15.3 Posting Information to the Bulletin Board Service 11-54

12. Resource Planning

12.1 Resource Planning Process 12-1
12.2 Resource Planning Workbench Utility 12-2
12.2.1 Create a Resource Reservation Request 12-3
12.2.2 Edit a Resource Reservation Request 12-6
12.2.3 Validate or Reject a Resource Reservation Request 12-7
12.2.4 Approve a Resource Reservation Request 12-7
12.2.5 Commit a Resource Reservation Request 12-8
12.2.6 Resource Planning Reports 12-8
12.2.7 Review Resource Timeline 12-9
12.2.8 Delete a Resource Reservation Request 12-9

12.3 Resource Definition Tool 12-10
12.3.1 Add a Resource 12-10
12.3.2 Modify a Resource 12-12
12.3.3 Delete a Resource 12-12
12.3.4 Synchronize Resource Listings 12-13

13. Production Planning

13.1 Production Request Editor 13-1
13.1.1 Launching the Production Request Editor 13-1
13.1.2 Create New Production Request 13-2
13.1.3 Edit/Modify Production Request 13-4
13.1.4 Review Data Production Requests 13-6
13.1.5 Reports 13-8
13.2 Production Planning Workbench 13-12
13.1.1 Launching the Planning Workbench and Planning Timeline 13-12
13.2.1 Create New Production Plan 13-13
13.2.2 Review Plan Timeline 13-15

14. Production Processing

14.1 Configure AutoSys 14-2
14.1.1 AutoSys Runtime Options 14-2
14.1.2 Configure Hardware Groups 14-4
14.2 Review Hardware Status 14-5
14.2.1 Review Hardware Status 14-5
14.2.2 Hardware Status View Options 14-6
14.3 Review DPR Dependencies 14-7
14.4 Review DPR Production Timeline 14-8
14.5 Modify Job Priority 14-9
14.6 Review Alarms 14-10
14.6.1 Review Alarms 14-10
14.6.2 Alarm Selection Configuration 14-11

14.7 Review Job Activities....14-12
14.7.1 Review Job Activities....14-12
14.7.2 Review Job Selection Criteria....14-13
14.8 Modify Job Status....14-14
14.9 Activity Log....14-16
14.10 Job Dependency Log....14-17
14.11 Defining Monitors/Browser....14-18
14.11.1 Defining Monitors/Browser....14-18
14.11.2 Monitor/Browser Reports....14-19
14.12 Database Maintenance Time Change....14-20
14.13 Production Reports....14-20
14.13.1 AutoSys Reports....14-21

15. Quality Assurance

15.1 Launching the QA Monitor GUI....15-1
15.2 DAAC Product QA....15-2
15.3 Product History....15-4
15.4 EOSView....15-5

16. Ingest

List of Figures

14.1-1. AutoSys Hardware Group File Example....14-4
14.9-1. Sample Activity Log....14-16
14.10-1. Sample Job Dependency Log....14-17
14.11-1. Sample Browser Screen....14-18

List of Tables

12.1-1. Resource Planning - Activity Checklist 12-2
12.2-1. Create a Resource Reservation - Quick-Step Procedures 12-4
12.2-2. Frequency Qualifiers 12-6
13.1-1. Production Request - Activity Checklist 13-1
13.1-2. Create New Production Request - Quick-Step Procedures 13-4
13.1-3. Edit/Modify Production Request - Quick-Step Procedures 13-6
13.1-4. Review Data Production Request - Quick-Step Procedures 13-8
13.1-5. Generate Reports - Quick-Step Procedures 13-11
13.1-6. Generate Reports - Quick-Step Procedures 13-11
13.2-1. Production Planning - Activity Checklist 13-12
13.2-2. Create New Production Plan - Quick-Step Procedures 13-15
13.2-3. Review Plan Timeline - Quick-Step Procedures 13-16
14.1-1. Production Planning - Activity Checklist 14-1
14.1-2. Runtime Options Table 14-2
14.13-1. Production Reports - Activity Checklist 14-21
14.13-2. AutoSys Reports - Quick-Step Procedures 14-22
15.1-1. Quality Assurance - Activity Checklist 15-1

11. SSI&T Operational Procedures

The SSI&T operational procedures are given in this section. They are organized by activity. The order in which the procedures appear loosely follows the order in which they will usually be performed.

These procedures present the use of GUIs supplied in Release A. Some procedures may have a command line equivalent; these are documented in the corresponding GUI help screens but are not presented here in the interest of simplicity.

A Note About the Order of Procedures

The science software I&T operational procedures contained within this document are ordered. The order is intended to *loosely* suggest a logical sequence which, when used as a "road map", represents an overall, sensible end-to-end SSI&T activity at the Release A DAACs. The ordering cannot, however, be interpreted as a detailed, step-by-step guide to SSI&T activities. In addition, since there are many factors that affect the actual SSI&T activities during Release A (e.g. Instrument Team deliveries, DAAC policies, agreements between the Instrument Teams and the DAACs, etc.), the ordering in this document can only be suggestive.

Many of the procedures outlined in this document are inter-related. A procedure may assume that another procedure has been completed. In general, the ordering of the procedures reflects this. The user should be aware, however, that this is not the case for all procedures. Therefore, depending on the SSI&T activity, the ordering suggested may not apply. Procedures may require other procedures that appear *after* the procedure requiring them.

Assumptions

All procedures in this paper assume the following: that the Instrument Team has delivered the science software to the DAAC and that Release A ECS is available at the DAAC.

Conventions

The following conventions are followed for explaining procedures:

Text that should be typed literally in the "Action" column of the procedures is displayed in `courier` font. Text within a literal command that represents a fill-in-the-blank object is displayed in *`italic courier`* font. (Example: `cd` *`mydir`* means type "cd" and then type the name of the correct directory.)

A command line in the "Action" column that should be typed in without a line break will be indicated by an indent in any following lines. The end of the command is indicated by <ENTER>, which stands for pressing the ENTER or RETURN key.

11.1 Preparation

11.1.1 Creating a New Earth Science Data Type (ESDT)

11.1.2 Validating Granule Level Metadata

11.1.3 Installing an Earth Science Data Type (ESDT) on the Data Server

11.1.4 Validating Successful ESDT Installation

11.1.5 Creating Metadata Configuration Files (MCF'

11.2 Acquiring the Delivered Algorithm Package (DAP)

11.2.1 Registering a Subscription for a DAP

11.2.2 Notification of DAP Arrival

11.2.3 Acquiring a DAP from Data Server

11.2.4 Acquiring a DAP via FTP

11.3 Configuration Management

11.3.1 Creating and Using a View in ClearCase

Purpose: To create a view in ClearCase. A view is necessary in order to make visible and accessible files and directories that have been checked into a VOB (Versioned Object Base). This procedure will also create a subdirectory where elements may be checked in.

Tools/Materials/Assumptions: ClearCase is available. A VOB has been created and its location is known. (Helpful hint: the ClearCase command to check what view, if any, is set is: `cleartool pwv <ENTER>`)

Note that this procedure should only be performed once for each account at each DAAC as part of the initial setup of the account.

Platform: SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Create a ClearCase view.	cleartool mkview tag *viewname* /net/mss1sun*xxxx*/data/viewstore/*viewname*.vws <ENTER>	*viewname* is the name of the view being created (usually the user name; example: jsmith) and *xxxx* are the initials of the DAAC being used (example: mss1sungsfc).
2	Set the ClearCase view to the one just created.	Cleartool setview *viewname* <ENTER>	*viewname* is the name of the view created in step 1.
3	Change directory to the VOB.	cd *VOBpathname* <ENTER>	The configuration management for SSI&T work has to be done in a Versioned Object Base (VOB). The DAAC System Administrator can explain which VOB to use. A list of all the available VOBs can be obtained by issuing the command: `cleartool lsvob <ENTER>` Try using the (default) VOB /ecs/*xxxx*_test, where *xxxx* is the DAAC initials (e.g. larc). To separate these files from files owned by other users, a subdirectory must be created.
4	Check out the current directory.	cleartool checkout nc . <ENTER>	This checks out the current directory from ClearCase. This is necessary because to add new elements (i.e. files) to ClearCase, the directory in which the elements are to be added must be checked out first; the directory itself is also being modified. `-nc` stands for “no comment” which means that ClearCase will not prompt for comments.
5	Create a subdirectory	cleartool mkdir -nc *dirname* <ENTER>	*dirname* is the name of the directory to be used with the newly created view. `-nc` stands for “no comment” which means that ClearCase will not prompt for comments. The directory that is created in this step will be automatically checked out as part of the `mkdir` command. This is the subdirectory in which the VOB administrator (i.e. System Administrator) will execute the ClearCase conversion script that is created in procedure 11.2.2 in order to import a whole directory structure into ClearCase.
6	Check in the subdirectory.	cleartool checkin nc dirname <ENTER>	*dirname* is the name of the directory created in step 5 that will be used with the newly created view. `-nc` stands for “no comment” which means that ClearCase will not prompt for comments.
7	Check in the current directory.	Cleartool checkin nc . <ENTER>	Here, the current directory (which is also in the VOB) is being checked in. `-nc` stands for “no comment” which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.3.2 Importing Multiple Files into ClearCase from a Directory Structure

Purpose: To put multiple files in a UNIX directory structure under CM using ClearCase.

Tools/Materials/Assumptions: The services of the VOB Administrator are required in order to complete this procedure. A VOB and a subdirectory are created to hold these files. No object files or executables exist in the source code directory. (Checking object files into ClearCase is discouraged. Object files and executables take up a lot of room and if they are checked into ClearCase they must be checked back out before source code can be recompiled.) The PGE was delivered and unpacked and/or untarred, creating a directory structure that contains the various types of files. The files are to be entered into ClearCase with the same directory structure as the delivery structure.

A ClearCase view does *not* have to be set to complete this procedure.

References: cleartool man clearcvt_unix, ClearCase User's Manual

Platform: SGI Power Challenge

The clearcvt_unix script must be run on SGI Power Challenge, and the conversion script cvt_script that clearcvt_unix creates *must* be run in the VOB on the SGI Power Challenge.

Steps:

#	Purpose	Action	Comment
1	Go to the parent directory of the UNIX directory structure to be brought into ClearCase.	cd *parentdir* <ENTER>	*parentdir* is the path of the directory that contains the directory structure to be brought into ClearCase. (This directory must *not* be in the VOB.)
2	Create a conversion script.	clearcvt_unix -r *dirname* <ENTER>	*dirname* is the name of the directory containing the directory structure. clearcvt_unix is a custom command that creates a script to check all the files in *dirname* into the VOB. -r ensures that any subdirectories below *dirname* will be recursively included in the script created.
3	Have the elements created and checked in by the System Administrator.	Ask the VOB Administrator to run the script "cvt_script" in the directory in the VOB where *dirname* should be checked into ClearCase. The System Administrator will log in, set a default view, move to the VOB directory, and run the master conversion script created in step 2.	The System Administrator is the only one who can run the cvt_script because it modifies the VOB. All VOB control is kept by the System Administrator for security purposes.

END OF PROCEDURE

11.3.3 Entering a Single File into ClearCase

Purpose: To put a single file under configuration management using ClearCase.

Tools/Materials/Assumptions: A Versioned Object Base (VOB) has been already mounted at the UNIX directory into which the file (element) is to be installed. A view has been created according to procedure 11.3.1. If a subdirectory *and* its files are to be checked in, use procedure 11.3.2 instead.

References: cleartool man pages, ClearCase User's Manual.

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Set the view.	cleartool setview *viewname* <ENTER>	*viewname* is the name of the view that was created in procedure 11.2.1.
2	Change directories to the directory in the VOB in which the file should be checked in.	cd *path* <ENTER>	*path* is the pathname to a subdirectory in the VOB.
3	Copy over the file to be checked in.	cp *path/filename* <ENTER>	*path* is the UNIX directory pathname where the file currently exists. This step places the file into your view, but not yet in the VOB.
4	Checkout the directory under ClearCase.	cleartool checkout nc . <ENTER>	. (a single period) stands for the current directory. This checks out the current directory from ClearCase. This is necessary because to add new elements to ClearCase, the directory in which the elements are to be added must be checked out first because the directory itself is also being modified. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
5	Create a new element under ClearCase.	cleartool mkelem -nc *filename* <ENTER>	*filename* is the name of the file being checked into ClearCase. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
6	Check in the file.	cleartool checkin -nc *filename* <ENTER>	*filename* is the name of the file being checked into ClearCase. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
7	Check in the directory.	cleartool checkin -nc . <ENTER>	. (a single period) stands for the current directory. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.3.4 Entering a New Directory into ClearCase

Purpose: To put a new (empty) directory under configuration management using ClearCase.

Tools/Materials/Assumptions: A Versioned Object Base (VOB) has been already mounted at the UNIX directory into which the directory (element) is to be installed. A view has been created according to procedure 11.3.1. If a subdirectory *and* its files are to be checked in, use procedure 11.3.2 instead.

References: cleartool man pages, ClearCase User's Manual.

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Set the ClearCase view.	cleartool setview *viewname* <ENTER>	*viewname* is the name of the view that was created in procedure 11.2.1.
2	Change directories to the directory in the VOB in which the new directory should be created.	`cd` *`path`* `<ENTER>`	*path* is the directory path of a parent directory in the VOB under which the new directory to be created.
3	Check out the parent directory, which is now the current directory.	`cleartool checkout -nc . <ENTER>`	The parent directory (current directory) is being modified by the addition of the new subdirectory, so it must be checked out first. Note that if someone else has already checked out the directory, permission to check it out will be denied. Contact the person to whom the directory is checked out. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
4	Create the new directory.	`cleartool mkdir -nc` *`directoryname`* `<ENTER>`	*directoryname* is the name of the new directory being created. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
5	Check in the new directory.	cleartool checkin -nc *directoryname* <ENTER>	*directoryname* is the name of the directory that was used as the path in step 4. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
6	Check in the parent directory.	cleartool checkin -nc . <ENTER>	. (a single period) stands for the current directory, which is the parent directory. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.3.5 Checking Out an Element from ClearCase

Purpose: To check out an element from ClearCase so that it can be modified.

Tools/Materials/Assumptions: The file or directory must be an element created in ClearCase, and the view should be configured to ensure the correct version of the file or directory is seen.

References: cleartool man pages for checkout, ClearCase User's Manual

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Set the ClearCase view.	cleartool setview *viewname* <ENTER>	*viewname* is the name of the view that was created in procedure 11.2.1.
2	Check out the file or directory.	`cleartool checkout -nc` *`element`* <ENTER>	*element* is a standard UNIX directory path name, ClearCase extended directory path name, or a file name. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
3	Cancel the checkout of the element if it is decided that no changes are to be made after all.	`cleartool uncheckout -nc` *`element`* <ENTER> Before canceling the checkout, ClearCase will prompt to choose whether to save a private copy of *element.*	*element* may be a standard UNIX directory path name, ClearCase extended directory path name, or a file name. Uncheckout is necessary because ClearCase will not allow an element to be checked in if no modifications have been made since checkout. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.3.6 Checking a Revised Element into ClearCase

Purpose: To check an element into ClearCase after modifying the element.

Tools/Materials/Assumptions: The file must be in a VOB and must have been checked out as a ClearCase element.

References: cleartool man pages for checkin, ClearCase User's manual.

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Set the ClearCase view.	cleartool setview *viewname* <ENTER>	*viewname* is the name of the view that was created in procedure 11.2.1.
2	Check in a revised file or directory.	`cleartool checkin -nc` *`element`* `<ENTER>`	*element* may be a standard UNIX directory path name, ClearCase extended directory path name, or a file name. -nc stands for "no comment", which means that ClearCase will not prompt for comments. If the element being checked in was not revised, ClearCase will not allow it to be checked back in. If this happens, complete step 3.
3	If the element cannot be checked back in because it was not revised, cancel the checkout of the element.	`cleartool uncheckout -nc` *`element`* `<ENTER>` Before canceling the checkout, ClearCase will prompt to choose whether to save a private copy of *element.*	*element* may be a standard UNIX directory path name, ClearCase extended directory path name, or a file name. ClearCase will not allow an element to be checked in if no modifications have been made since checkout. Use the `-rm` option to disable the prompting to make a private copy of the element. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.3.7 Applying Labels to ClearCase Elements

11.4 SSIT Manager GUI

11.4.1 General Setup of the SSIT Manager

The SSIT Manager is the top-level graphical user interface (GUI) environment presented to personnel involved in performing SSI&T. Its purpose is to bring together the tools needed for SSI&T into a single, graphical environment. The SSIT Manager runs only on the Release A AIT Suns.

It is recommended that the xterm window used to start the SSIT Manager not be used for any other purpose. The reason is that messages from the SSIT Manager will appear in this window from time to time; the user may want to iconize (minimize) this xterm window to keep the desktop more manageable.

To run the SSIT Manager, some initial setup and configuration is necessary. Detailed procedures for configuring the SSIT Manager is contained in sections 11.4.2 and 11.4.3.

Note that in many regards, the SSIT Manager software was developed in the same manner as science software delivered to the DAACs by the Instrument Teams: It uses the SDP Toolkit internally; it uses a Process Control File; and it requires that certain environment variables point to locations in the resident SDP Toolkit directory.

Minimum Setup

The SSIT Manager can only be run on the AIT Suns. Currently, the SSIT Manager setup files assume that the C shell or the more advanced T shell is the current shell. The following environment variables must be set. The commands shown below may be entered at the UNIX shell prompt each time the SSIT Manager is to be run; however, it is most convenient to place them into a login startup script such as $HOME/.cshrc.

DPATMGR_HOME

The environment variable DPATMGR_HOME must be set on the AIT Sun to the home directory of the SSIT Manager software. Example:

```
setenv DPATMGR_HOME manager_home_dir
```

where *manager_home_dir* is the home directory on the AIT Sun of the SSIT Manager.

PGS_PC_INFO_FILE

The environment variable PGS_PC_INFO_FILE must be set on the AIT Sun to the Process Control File (PCF) used by the SSIT Manager. This PCF should not be confused with the PCFs used by science software run on the Science Processor (SGI). When installed, the location of the default PCF used by the SSIT Manager is $DPATMGR_HOME/data/PCF.v5.ssit.*machinename* where *machinename* is the name of the machine on which the SSIT Manager was installed. At installation time, PGS_PC_INFO_FILE is set to this PCF.

Typically, users will want to customize this PCF. The most common customization is to change the checklist file pointed to by the SSIT Manager to one customized by the SSI&T personnel (refer to section 11.3.2 which addresses this in more detail). The pointer to the checklist file is in this PCF and hence, the PCF must be customized.

To customize the PCF used by the SSIT Manager, first make a private copy of it, for example, in your home directory. Changes should be made to this private copy. The environment variable PGS_PC_INFO_FILE should then be reset to point to your private PCF rather than the default PCF, as in:

```
setenv PGS_PC_INFO_FILE $HOME/myPCF
```

where $HOME/myPCF is the full pathname of your private copy of the PCF to be used with the SSIT Manager when you run it.

PGSHOME

The environment variable PGSHOME must be set to the location of the SDP Toolkit home directory on the AIT Sun. Example:

```
setenv PGSHOME sdp_home_dir
```

where *sdp_home_dir* is the location of the SDP Toolkit home directory on the AIT Sun.

PGSMSG

The environment variable PGSMSG must be set on the AIT Sun to the location of the SDP Toolkit message directory. If PGSHOME has already been set (see above), PGSMSG can be set as:

```
setenv PGSMSG $PGSHOME/message
```

Other Environment Variables

The SSIT Manager requires many other environment variables to be set. To set all of them, just source the following file on the AIT Sun:

```
source $DPATMGR_HOME/bin/sun5/DpAtEnv.csh
```

Note that this file is set up to work only in the C shell (or T shell).

DISPLAY

The SSIT Manager needs to know where to display itself. If the user is on one of the AIT Suns, the SSIT Manager will by default get displayed on that AIT Sun. In this case, the environment variable DISPLAY will be set to this machine automatically. If, however, the user is running the SSIT Manager from another machine (for example, remotely from a NCD X Terminal), the environment variable DISPLAY must be set to that machine by the user. Example:

```
setenv DISPLAY machinename:0.0
```

where *machinename* is the machine (IP address or fully-qualified domain name) on which the SSIT Manager is to be displayed.

If access to the SSIT Manager will usually be from the same machine, it may prove convenient to include this setting in a login startup script, such as $HOME/.cshrc. If this is not the case, however, it may be more practical to set it at the UNIX shell prompt each time the SSIT Manager is to be run.

Example Minimum Setup Script

The following is a fragment of a .cshrc file (in the user's home directory) used at the Goddard DAAC illustrating the minimum setup of the SSIT Manager discussed in the sections above:

```
setenv DPATMGR_HOME /RelA_IT/AI_T/
```

```
setenv PGS_PC_INFO_FILE $HOME/myPCF
setenv PGSHOME /data/RelA/AI_T/toolkit/PGSTK
setenv PGSMSG $PGSHOME/message
source $DPATMGR_HOME/bin/sun5/DpAtEnv.csh
```

END OF PROCEDURE

11.4.2 Setup of Checklist for SSIT Manager

Purpose: To prepare a checklist containing all the operational procedures for science software integration and test.

The SSIT Manager GUI requires a checklist which it can access. The checklist of operational procedures must first be prepared in a UNIX session using any available editor, such as emacs, vi, or xedit. The SSIT Manager must then be linked to the checklist before it is invoked.

Tools/Materials/Assumptions: This procedure assumes that an account and the proper environment has been setup for the operations staff to test PGEs and a specific PGE is to be delivered for testing. The following environmental variables should be set in login files, as explained in below: DPATMGR_HOME, which points to the SSIT Manager directory; the environmental variable PGSHOME, which points to the SDP toolkit for the SSIT Manager; and the environmental variable PGS_PC_INFO_FILE, which points to the PCF in the directory in which the SSIT Manager will be used (usually, $HOME).

References: Release A Maintenance and Operations Procedures, 609-CD-XXX-XXX

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Start an xterm session on the Sun in order to create a checklist file to be used by the SSIT Manager GUI.	`xterm & <ENTER>`	
2	Copy the sample (default) checklist to the directory from which the SSIT Manager will be run.	`cp $DPATMGR_HOME/data/checklist.sample $HOME/`*`mychecklist`* `<ENTER>`	*$HOME* is the user's home directory and is the usual directory from which the SSIT Manager will be run. *mychecklist* is the name of the customized checklist, such as "MODIS_checklist."

Steps: (continued)

#	Purpose	Action	Comment
3	Edit the checklist using any available editor.	`xedit` *`mychecklist`* `<ENTER>`	Available editors include vi, emacs and xedit. m*ychecklist* is the user-selected name for the checklist file.
4	Define the path for the SSIT Manager's database and a title which will appear above the checklist in the SSIT Manager GUI.	Edit the lines containing: `DATABASE=`*`ssitmgrusedir/filename`* `CHECKLIST=`*`title`*	*ssitmgrusedir* is the complete path from which the SSIT Manager will be run; usually, this is the user's home directory. *filename* is the base name for the database files that will be created by the SSIT Manager. f*ilename* can be any name. When the SSIT Manager is run, two database files will be created: *filename.*`dir` and *filename.*`pag`. (These files are NOT Sybase files.) *title* is a user-selected name which will appear in the SSIT Manager GUI above the checklist items.
5	Edit and add items for the checklist. These items are the text that specifies the SSI&T steps to perform.	`ITEM=`*`proceduralstep1`* `ITEM=`*`proceduralstep2`* ...	*procedural-steps* are entered as text strings (without quotes). The items form a list of operational procedures to be followed and checked off when completed. These steps can possibly be derived from existing documentation; a good reference is the agreement between the DAAC and the IT. Add ITEM lines as needed.
6	Save the checklist file and quit the editor.		
7	If the database files already exist in the directory specified by the DATABASE variable with the same *filename*, then delete them.	`rm` *`filename`*`.dir` `<ENTER>` `rm` *`filename`*`.pag` `<ENTER>`	*filename* is the name that was used in the DATABASE variable in the checklist in step 4 If the .dir and .pag files exist, and the SSIT Manager is invoked, an error will occur. Existing .dir and .pag files must be deleted.

END OF PROCEDURE

11.4.3 Running the SSIT Manager for the First Time

Purpose: To configure the SSIT Manager.

Tools/Materials/Assumptions: The SSIT Manager has been installed correctly on the Sun. An account exists for the user on the Sun. The current shell being used is the C shell (*csh*) or the T shell (*tcsh*) and a C shell configuration file named .cshrc exists in the home directory. The location of the SSIT Manager's home directory is known. The user has an environment variable, HOME, set to the user's home directory. The environment variable DPATMGR_HOME has been set to the path for the home directory of the SSIT Manager. The environment variable DISPLAY is set to the name or IP address of the terminal being used.

Refer to section 11.3.1 for a general discussion of the SSIT Manager setup.

References:

Release A Maintenance and Operations Procedures, 609-CD-001-001

Help screen available through SSIT Manager (from SSIT Manager, choose **Help**)

README file, $DPATMGR_HOME/DpAtREADME.USER

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Make sure the environmental variables PGSHOME and PGSMSG are correctly set.	`echo $PGSHOME <ENTER>` `echo $PGSMSG <ENTER>`	If these variables are not set, see the M&O contact; the account is not properly configured.
2	Make sure the environmental variable DPATMGR_HOME has been correctly set.	echo $DPATMGR_HOME <ENTER>	If this variable is not set, see the M&O contact; the account is not properly configured.
3	Copy the SSIT Manager's Process Control File (PCF) to the directory from which the SSIT Manager will be run. Typically, this will be the user's home directory.	cp $DPATMGR_HOME/data/PCF.v5.ssit.*hostname* $HOME/*myPCF* <ENTER>	DPATMGR_HOME is the environment variable set to the SSIT Manager's home directory (see section 11.3.1), *hostname* is the name of the machine running the SSIT Manager (example: dps3sunedf), and *myPCF* is any desired filename for the copy being made. It is this copy that will be edited in the steps below.
4	Edit the copied PCF using any available editor.	`xedit $HOME/`*`myPCF`* `<ENTER>`	*myPCF* stands for the copy of the PCF created in step 3.
5	Search the copied PCF for the identifier 603 which begins the line containing the location of the checklist file and insert the path to the checklist.	Search for 603 and edit this line: 603|DpCAtMgrLogDatabaseInit|*checklistpath/mychecklist*	*checklistpath* is the location of the user's checklist file. *mychecklist* is the name of the user's checklist file that was created in procedure 11.3.2.
6	Save the file and quit the editor.		
7	Reset the environment variable for the Process Control File.	setenv PGS_PC_INFO_FILE $HOME/*myPCF* <ENTER>	*myPCF* refers to the local copy of the SSIT Manager's PCF created in step 3. This command should be placed in a login startup file so that it is properly set at future logins.

Steps: (continued)

#	Purpose	Action	Comment
8	If the SSIT Manager will access files from a VOB, set the ClearCase view.	`cleartool setview` *`viewname`* `<ENTER>`	*viewname* is the name of the view that was created in procedure 11.2.1.
9	Run the SSIT Manager.	`DpAtMgr & <ENTER>`	The ampersand causes the execution to be carried out as a background process. Various messages from the SSIT Manager will appear in this window as it is running. It is advised that no other activities be done within this window until termination of the SSIT Manager.

END OF PROCEDURE

11.4.4 Routine Running of the SSIT Manager

Purpose: To run the SSIT Manager once it has been configured.

Tools/Materials/Assumptions: The SSIT Manager has been installed correctly on the Sun. An account exists for the user on the Sun. The current shell being used is the C shell (*csh*) or the T shell (*tcsh*) and a C shell configuration file named .cshrc exists in the home directory. The location of the SSIT Manager's home directory is known. The user has an environment variable, HOME, set to the user's home directory. The environment variable DPATMGR_HOME has been set to the path for the home directory of the SSIT Manager. The environment variable DISPLAY is set to the name or IP address of the terminal being used. The SSIT Manager was properly set up according to procedures in sections 11.4.1, 11.4.2, and 11.4.3.

References:

Release A Maintenance and Operations Procedures, 609-CD-001-001

Help screen available through SSIT Manager (from SSIT Manager, choose **Help**)

README file, $DPATMGR_HOME/DpAtREADME.USER

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Log into the Sun machine or the machine typically used for SSI&T.		
2	Set the DISPLAY environment variable (if not already set automatically in a login startup script).	`setenv DISPLAY` *`hostname`*`:0 <ENTER>`	*hostname* is the name or IP address of the terminal being used. Example: setenv DISPLAY ncd2:0
3	If the SSIT Manager needs to access files from a ClearCase VOB, set the ClearCase view.	`cleartool setview` *`viewname`* `<ENTER>`	*viewname* is the name of the view that was created in procedure 11.2.1.
4	Run the SSIT Manager.	`DpAtMgr & <ENTER>`	The ampersand (&) causes the execution to be carried out as a background process. Various messages from the SSIT Manager will appear in this window as it is running. It is advised that no other activities be done within this window until termination of the SSIT Manager.

5	Optionally dump and view the contents of all entries made to all items in the SSIT Manager Checklist Log.	To view the SSIT Manager Log contents on screen, open a new xterm window on the same Sun that is running the SSIT Manager and enter:
		`DpAtMgrLogDump` *`databasename`* `<ENTER>`
		To redirect the output to a file:
		`DpAtMgrLogDump` *`databasename`* `>` *`outputfile`* `<ENTER>`
			`databasename` is the name of the database that is specified by directory path and filename in the `DATABASE` line of the checklist file being used.
			`mychecklist` is the SSIT Manager checklist created in procedure 11.3.2.
			The SSIT Manager Log database is not editable directly and utilizes the UNIX ndbm flat-file database utility. It has no connection to other SSI&T databases such as the Sybase used for storing PGE and DPR information.

END OF PROCEDURE

11.5 Standards Checking

11.5.1 Checking ESDIS Standards Compliance: FORTRAN 77

Purpose: To check that FORTRAN 77 language source files are compliant with the ESDIS *Data Production Software and Science Computing Facility (SCF) Standards and Guidelines* document. The current version of this document requires all FORTRAN 77 code to be compliant with the ANSI FORTRAN 77 with some extensions. The COTS used for this task is FORCHECK.

Tools/Materials/Assumptions: The FORTRAN 77 source files are available, accessible, and have read permissions. The environment variable FCKCNF has been set to the location of the FORCHECK configuration file. FORCHECK is run from the Sun that has the FORCHECK license. The Status Message Facility (SMF) files have been compiled (see section 11.6.2).

FORCHECK can be started from the SSIT Manager; however, this procedure instead assumes that FORCHECK will be run from the command line using a script. The form of this script is shown below:

FORCHECK run script:

```
forchk -l listfile.lst \
-I :incdir1:incdir2:[ ... ] \
/path/source_code1.f \
/path/source_code2.f \
          .
          .
          .
/path/source_code3.f
```

References:

Release A Maintenance and Operations Procedures, 609-CD-001-001

FORCHECK for Sun/SunOS, A Fortran Verifier and Programming Aid, Installation Guide

FORCHECK for Sun/SunOS, A Fortran Verifier and Programming Aid, User Guide

Help screen available through SSIT Manager (from SSIT Manager, choose Help)

ANSI FORTRAN 77 Programming Language Standard, X3.9-1978

IEEE Standard 1003.9, POSIX FORTRAN77, Language Interfaces, Part 1: Binding for System Application Program Interface (API)

Data Production Software and Science Computing Facility (SCF) Standards and Guidelines, January 14, 1995 (CH-01 2/15/95), also on World Wide Web:

URL: http://spsosun.gsfc.nasa.gov/SCFStand.html

Platform: Sun (FORCHECK is *only* available on ait1sun; please be certain that is the Sun being used)

Steps:

#	Purpose	Action	Comment

1	Create a script file that contains the command lines to run FORCHECK.	Using editor of choice, create a file *fc_runfile* using the FORCHECK run script template above where *fc_runfile* is the filename.
		The first line of fc_runfile should only contain the line:
		`forchk -l listfile.lst \`
		where *listfile* is replaced with the filename for the list output that will be produced. The extension must be *.lst*.
		The second line of fc_runfile should only contain the line:
		`-I :incdir1:incdir2: ... \`
		where each *incdirN* is replaced with the full pathname of the location of the include files used by the software. Typically, the include directories of the SDP Toolkit , HDF, and the PGE will need to be listed. Add other directories as needed. Separate each pathname with a colon only. The entire line should be contained on a single line only with no newline characters embedded. A backslash should be the last character on this line.
		Each of the lines that follow in *fc_runfile* should contain the full pathname of each of the FORTRAN 77 source files. Place one pathname per line. Terminate each line except the last with a backslash.
			The last character of each line in *fc_runfile* must be the backslash character (\) except for the last line of the file.
			A colon by itself should be placed at the beginning of the list. This causes the current directory to be in the list of directories searched for include files.
			The last FORTRAN 77 source file listed in *fc_runfile* should not be terminated with a backslash.

2	Save the FORCHECK run script file and exit the editor.		
3	Give the FORCHECK run script file execute permissions.	`chmod +x fc_runfile` <ENTER>	*fc_runfile* is the name of the file created in step 1.
4	If necessary, set the ClearCase view.	`cleartool setview viewname` <ENTER>	*viewname* is the name of the view created in procedure 11.2.1.

Steps: (continued)

#	Purpose	Action	Comment
5	Run the FORCHECK run script file, saving the output.	`fc_runfile >& fc_output` <ENTER>	*fc_output* is the filename of the output file from FORCHECK.
6	Examine the output file and the list file.	Using desired editor, view the contents of the output, *fc_output* and the list file, *listfile.lst*.	The output file contains the errors, warnings and messages from FORCHECK along with a useful summary at the bottom. The list file contains the source code along with embedded messages from FORCHECK.

END OF PROCEDURE

11.5.2 Checking ESDIS Standards Compliance: Fortran 90

Purpose: To check that Fortran 90 language source files are compliant with the ESDIS *Data Production Software and Science Computing Facility (SCF) Standards and Guidelines* document.

Tools/Materials/Assumptions:

References:

Platform: SGI Power Challenge

Steps: In the following, the assumption is that standards compliance checking is performed as a separate process from the build process.

#	Purpose	Action	Comment

END OF PROCEDURE

11.5.3 Checking ESDIS Standards Compliance: C

Purpose: To check that C language source files are compliant with the ESDIS *Data Production Software and Science Computing Facility (SCF) Standards and Guidelines* document. The current version of this document required all delivered C source code to be ANSI compliant.

Tools/Materials/Assumptions: The C source files are available in a single directory, accessible, and have read permissions. Any non-standard include files are available in known directories, accessible, and have read permissions. The C compiler is accessible on the Science Processor (SGI Power Challenge). The current shell being used is the C shell (*csh*) or the T shell (*tcsh*). The Status Message Facility (SMF) files have been compiled (see procedure 11.5.2). Compliance checking for ESDIS standards is done via an SGI Power Challenge C compiler flag. Checking may be done as part of the science software build process or as a separate, preliminary step.

References:

cc man pages on the SGI Power Challenge

SDP Toolkit User's Guide for the ECS Project, 333-CD-003-002

The Toolkit Primer, available through the World Wide Web:

URL: http://newsroom.hitc.com/sdptoolkit/primer/tkprimer.html

ANSI C Programming Language Standard, X3.159-1989

Data Production Software and Science Computing Facility (SCF) Standards and Guidelines, January 14, 1995 (CH-01 2/15/95), also on World Wide Web:

URL: http://spsosun.gsfc.nasa.gov/SCFStand.html

Platform: SGI Power Challenge

Steps: In the following, the assumption is that compliance checking is performed as a separate process from the build process.

#	Purpose	Action	Comment
1	Start a new shell on the SGI Power Challenge science processor within which to do the checking.	Either: From the SSIT Manager GUI, choose the Tools menu. Then choose xterm. Then telnet into the SGI Power Challenge. or In any currently available xterm window, spawn a new xterm session with: `xterm & <ENTER>` Then telnet into the SGI Power Challenge.	Science software will be run on the SGI Power Challenge. Therefore, checking for ANSI compliance via the compiler should be done on the same machine.
2	Setup the toolkit environment.	`setenv PGSHOME` *`toolkithomedir`* `<ENTER>` `source $PGSHOME/bin/`*`sgiX`*`/pgs-dev-env.csh <ENTER>`	*toolkithomedir* is the home directory of the desired toolkit (see section 11.4.3). *X* in *sgiX* refers to the appropriate processor (see section 11.4.3 for details). Example: `source $PGSHOME/bin/sgi32/pgs-dev-env.csh <ENTER>`
3	If necessary, set the ClearCase view.	`cleartool setview` *`viewname`* `<ENTER>`	*viewname* is the name of the view created in procedure 11.1.2.
4	Go into the directory containing the source files.	`cd` *`srcdirectory`* `<ENTER>`	*srcdirectory* is the full path to the location of the source files being checked. If the source files have been checked into ClearCase, this directory will be in the VOB.

Steps: (continued)

#	Purpose	Action	Comment
5	Run the C compiler with ANSI checking enabled. Save the results to an output report file.	Within the xterm window, enter: `cc -c -32 -ansi [-I$PGSINC] [-I$HDFINC] [[-IOtherInclFiles]...] sourcefiles >& reportfile` <ENTER> The terms in brackets (*[ ]*) are used to optionally specify locations of include files: $PGSINC for SDP Toolkit include directory, $HDFINC for HDF include directory, and *OtherInclFiles* represents one or more include directories necessary to compile the software. These include files should be described in the documentation that came with the PGE.	*sourcefiles* is a list of C source files or a wildcard template and *reportfile* is the filename under which to save the results. The `-c` flag causes only compilation and no linking. The `-ansi` is the SGI Power Challenge C compiler flag to enable ANSI checking. The `-32` flag causes compilation in old 32-bit (IRIX 5.3) mode (`-n32` and `-64` are used for new 32-bit and 64-bit modes (IRIX 6.X), respectively). Each `-I` flag precedes the name of a directory in which to search for non-standard include files. Apply the terms in brackets only as necessary. Do not type the brackets on the actual command line. Do not use the `-I` option for include files that are in the standard directories or in the current directory. Example: `cc -c -32 -ansi -I$PGSINC -I$HDFINC -I/ecs/gsfc_test/modis/include *.c >& reportfile` Note: if *reportfile* exists, it will *not* be overwritten and the command will fail. Delete the existing *reportfile* or use a different name.
6	Review the report file for any non compliance code.	Visually scan the file using `more` or `cat`, or view the file using a preferred text editor (e.g. `vi`, `emacs`, `xedit`).	Resolve any issues of non-compliance according to prescribed procedures in the DAAC-IT agreement.

END OF PROCEDURE

11.5.4 Checking ESDIS Standards Compliance: Ada

Purpose: To check that Ada language source files are compliant with the ESDIS *Data Production Software and Science Computing Facility (SCF) Standards and Guidelines* document.

Tools/Materials/Assumptions:

References:

Platform: SGI Power Challenge

Steps:

Since Ada compliance checking is implicit in the compiler, these steps can be entirely ignored. The Ada software build process will reveal any non-compliant code.

#	Purpose	Action	Comment

END OF PROCEDURE

11.5.5 Prohibited Function Checker

Title: Prohibited Function Checker

Purpose: To check C, FORTRAN 77, Fortran 90, and Ada language source files for the occurrence of functions that are prohibited in the ECS DAAC production environment.

Tools/Materials/Assumptions: The SSIT Manager is running. The source file(s) are available, accessible, and have read permissions. The formatted text (ASCII) files containing the list of prohibited functions exist in the directory stored in the environment variable DPATMGR_DAT. Their names are:

prohibitedFunctionsAda.txt

prohibitedFunctions.C++.txt

prohibitedFunctions.C.txt

prohibitedFunctions.F77.txt

prohibitedFunctions.F90.txt

These files are installed with the SSIT Manager. They may be edited. Directions for editing are contained within each file.

If the source code files to be checked are in a VOB in ClearCase, a view has been set before the SSIT Manager was started.

References:

Release A Maintenance and Operations Procedures, 609-CD-001-001

Help screen available through SSIT Manager (from SSIT Manager, choose **Help**)

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Invoke Prohibited Function Checker.	From Tools menu of the SSIT Manager choose Standards Checkers. Then choose Prohibited Function Checker.	The Prohibited Function Checker GUI will be displayed.
2	Choose the source files to check.	Click on the button labeled Analyze.	The File Selector GUI will be displayed.
3	Choose directory containing the source files.	Within the Directories subwindow, double-click on the desired directory. Repeat until the directory with the source files is displayed.	The files contained within the directory will be displayed in the Files subwindow.
4	Choose the file(s) to check.	Within the Files subwindow highlight the files to check. Choose groups of contiguous files by holding down and dragging the left mouse button. Choose non-contiguous files by holding down the Control key while clicking on the filenames.	The selected files will be displayed in the Selections subwindow. Note that files can only be checked from one directory at a time. Choosing another directory causes any filenames listed in this subwindow to disappear.
5	Run the checker.	Click on the button labeled Ok.	The File Selector GUI will disappear. The files chosen will be displayed in the Prohibited Function Checker window as they are checked.
6	Check the results.	Click on the button labeled Report.	The Report GUI will be displayed. For each file, a list of prohibited functions found will be displayed.
7	Optionally, save the results to a file or send to printer.	To save the report to a file, click on the button labeled Save. To print the report, click on the button labeled Print.	Choosing Save To File will bring up a Save GUI. Use this GUI to specify directory and filename under which to save the results. Choosing Print will send the results to the default printer.
8	Optionally, view the occurrences of prohibited functions in the source files.	In the Prohibited Function Checker GUI, highlight one of the source files listed. Then click on the button labeled View.	The Source Code GUI will be displayed. Occurrences of prohibited functions found will be highlighted. The button labeled Next will bring successive occurrences of highlighted prohibited functions into view. (The Next button does not bring in the next file.) The Done button closes the Source Code GUI.
9	Quit the Prohibited Function Checker.	In the Prohibited Function Checker GUI, click on the button labeled Quit.	The Prohibited Function Checker GUI will disappear.

END OF PROCEDURE

11.5.6 Checking Process Control Files

Purpose: To check that Process Control Files (PCFs) are syntactically correct and contain all necessary information for PGEs to run within the ECS DAAC production environment.

Tools/Materials/Assumptions: The SSIT Manager is running. The Process Control File(s) are available, accessible, and have read permissions.

If the source code files to be checked are in a VOB in ClearCase, a view has been set before the SSIT Manager was started.

References:

Release A Maintenance and Operations Procedures, 609-CD-001-001

Help screen available through SSIT Manager (from SSIT Manager, choose **Help**)

Steps:

#	Purpose	Action	Comment
1	Invoke Process Control File (PCF) Checker.	From Tools menu, choose Standards Checkers. Then choose Process Control File Checker.	The PCF Checker GUI will be displayed.
2	Choose directory containing the PCF(s).	Within the Directories subwindow, double-click on the desired directory. Repeat until the proper directory is displayed.	The files contained within the directory will be displayed in the Files subwindow according to the Filter defined in the Filter subwindow.
3	Choose the PCF to check.	Within the Files subwindow, highlight the PCF to check.	Only one PCF can be checked at a time.
4	Run the checker.	Click on the button marked Check PCF.	PCF Checker Results GUI will be displayed. Errors and warnings found will be listed in the main subwindow.
5	Optionally, save the results to a file or send to printer.	To save the report to a file, click on the button labeled Save. To print the report, click on the button labeled Print.	Choosing Save To File will bring up a Save GUI. Use this GUI to specify directory and filename under which to save the results. Choosing Print will bring up a Working GUI. Choosing the OK button will send the results to the default printer.
6	Optionally, check another PCF or quit.	To check another PCF, click on Check Another. To quit, click on Quit.	If checking another PCF, repeat steps 2 through 5. If quitting, the GUI will disappear.

END OF PROCEDURE

11.5.7 Extracting Prologs

11.6 Compiling and Linking

11.6.1 Updating the Process Control File (PCF)≠

Purpose: To prepare the Process Control File (PCF) for the DAAC environment.

Tools/Materials/Assumptions: A PCF was included in the delivery package and it worked (as is) in the SCF environment. This PCF was based on the template PCF included with SDP Toolkit 5 (in $PGSHOME/runtime/PCF.v5). The directory structure and locations of necessary input and output files in the DAAC environment are known.

References:

SDP Toolkit User's Guide for the ECS Project, 333-CD-003-002

The Toolkit Primer, available through the World Wide Web,

URL: http://newsroom.hitc.com/sdptoolkit/primer/tkprimer.html

Process Control File template, current version on the Release A machines

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	If the PCF was checked into ClearCase, set the ClearCase view.	`cleartool setview` *`viewname`* `<ENTER>`	*viewname* is the name of the view created in procedure 11.2.1.
2	Change directory to the directory containing the PCF file; if ClearCase is being used, this will be in the VOB.	`cd` *`path`* `<ENTER>`	*path* is the directory path of the directory containing the PCF. If ClearCase is being used for CM, this directory will be in the VOB.
3	Run the Process Control File Checker on the PCF.	See procedure 11.4.6 for details. If any errors are detected, fix them.	This ensures that the PCF is syntactically correct before it is modified and contains no detectable errors.
4	If the PCF has been checked into ClearCase, then check it out first.	`cleartool checkout -nc` *`PCFfilename`* `<ENTER>`	*PCFfilename* is the filename of the PCF. See procedure11.2.5 for details.
5	Edit the PCF.	Use any desired text editor (`vi`, `emacs`, `xedit`, etc.).	

Steps: (continued)

#	Purpose	Action	Comment
6	Where necessary, make changes to the default directories specified in each section of the PCF. All paths specified in the PCF must exist on the SGI Power Challenge at the DAAC.	Each section begins with a line consisting of a ? in the first column followed by a label: ? `PRODUCT INPUT FILES` ? `PRODUCT OUTPUT FILES` ? `SUPPORT INPUT FILES` ? `SUPPORT OUTPUT FILES` ? `INTERMEDIATE INPUT` ? `INTERMEDIATE OUTPUT` ? `TEMPORARY I/O` Each of these section heading lines will then be followed (not necessarily immediately; comment lines may be in between) by a line that begins with a `!` in the first column. If it reads: `! ~/runtime` leave it unchanged. If another path is listed instead of ~/runtime, it will probably need to be changed to a path that exists on the SGI Power Challenge at the DAAC. This path refers to the default directory in which to look for the files of that section when a path in that entry is not specified.	Use an absolute path, not a relative one. The tilde (~) is a symbol for PGSHOME. When specifying a default directory other than `~/runtime`, do not use a tilde. The science software may be using these environment variables to set the default paths; one such variable is available for each section. This step changes the settings of these variables to match the environment of the DAAC.
7	Look for science software specific entries in each section and make changes to paths (field 3) as necessary. All paths specified in the PCF must exist on the SGI Power Challenge at the DAAC. The science software specific entries will have logical IDs *outside* the range 10,000 - 10,999 (in the first field).	Where necessary, replace the paths in the third field of each entry with the paths appropriate for the current environment. These should be paths that exist on the SGI Power Challenge at the DAAC. In general, do not alter file entries that are used by the Toolkit itself. These have logical ID in the range 10,000 - 10,999. Exceptions to this follow in steps 7 - 9.	Example: in the following line, `100|L1A.granule|/MODIS/run/input|||1` change `/MODIS/run/input` (field 3) to the appropriate path in the DAAC where the input file L1A.granule is stored. Use an absolute path, not a relative one. Do not include the filename with the path name.

Steps: (continued)

#	Purpose	Action	Comment
8	Add a necessary entry to the SUPPORT OUTPUT FILES section. It is missing from the PCF template that went out with the SDP Toolkit 5. Add it here.	Add the following line to the SUPPORT OUTPUT FILES section: `10113|eventLogger.log |||||1`	Place the entry exactly as shown beginning in column 1 (note that it should be typed on a single line; it is shown on two lines here due to lack of space). It can be placed anywhere among the other entries already listed in the SUPPORT OUTPUT FILES section.
9	Verify that the SUPPORT OUTPUT FILES section contains an entry to the shared memory pointer file.	In the SUPPORT OUTPUT FILES section, look for the line: `10111|ShmMem|~/runtime||||1` If it is not within this section, add it.	This line should be in the SUPPORT OUTPUT FILES section exactly as shown beginning in column 1 (note that it should be typed on a single line; it is shown on two lines here due to lack of space). Even though this entry was included in the PCF template, it may have been left out of the science software delivery under the (incorrect) assumption that it is not needed when shared memory is not requested by the PGE.
10	Verify that the USER DEFINED RUNTIME PARAMETERS section contains an entry for 10112.	In the USER DEFINED RUNTIME PARAMETERS section, look for the line: `10112|`*`comment`*`|1` If it is not within this section, add it.	*comment* may be any string except an empty string. A suggested comment is: `Event logging is turned on` This parameter, 10112, turns event logging on and off. (In step 9 the event log file name was specified.) The "1" in the third field indicates that event logging is turned on.

Steps: (continued)

#	Purpose	Action	Comment

11	The location of certain database files pointed to in the PCF may need to be changed. Look for these entries and change them as appropriate.	In the following entries:
		`10301|leapsec.dat|~/lib/database/TD||||1`
		`10401|utcpole.data|~/lib/database/CSC||||1`
		`10402|earthfigure.dat|~/lib/database/CSC||||1`
		`10501|.|~/lib/database/EPH||||1`
		`10601|de200.eos|~/lib/database/CBP||||1`
		replace the `~/lib/database` part of each path (third field) with:
		`~/database/`*`mode`*
		where Example for entry 10301 for the 64-bit compiler on the SGI Power Challenge, change it to
		`10301|leapsec.dat|~/database/sgi64/TD||||1`
			mode is one of: `sgi`, `sgi32`, `sgi64`, depending on which version of the SDP Toolkit is being used (see section 11.5.3 for details of SDP Toolkit versions).
			The PCF template delivered with SDP Toolkit 5.1 assumed a different directory structure for these database files.
			Other PCF entries may be affected than those listed to the left. Change all that have `~/lib/database` as part of the path.
			Note that each entry shown in the middle column should actually be on one line. They are shown here as wrapped to fit within the column.

12	Rerun the Process Control File Checker on the revised PCF.	See procedure 11.4.6 for details. If any errors are detected, fix them.	This ensures that no new errors were introduced into the PCF.
13	If the PCF had been checked out of ClearCase, then check the revised version back in.	`cleartool checkin -nc` *`PCFfilename`* `<ENTER>`	*PCFfilename* is the filename of the PCF. See procedure11.2.6 for details. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE

11.6.2 Compiling Status Message Facility (SMF) Files

Purpose: To compile Status Message Facility files (SMF files) delivered with science software into message files and include (header) files. These files will be used by the science software during runtime.

Tools/Materials/Assumptions: The delivered SMFs are available, accessible, and have read permissions. The program *smfcompile* (included with the Toolkit) is accessible.

References:

SDP Toolkit User's Guide for the ECS Project, 333-CD-003-002

The Toolkit Primer, available through the World Wide Web,

URL: http://newsroom.hitc.com/sdptoolkit/primer/tkprimer.html

Platform: SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	Start a new shell on the SGI Power Challenge science processor within which to run smfcompile.	Either: From the SSIT Manager GUI, choose the Tools menu. Then choose xterm. Then telnet into the SGI Power Challenge within the xterm. or In any currently available xterm window, spawn a new xterm session with: `xterm & <ENTER>` Then telnet into the SGI Power Challenge within the xterm.	Science software will be run on the SGI Power Challenge. Therefore, compiling status message files should be done on the same machine.
2	If ClearCase is being used for CM, set the ClearCase view.	`cleartool setview` *`viewname`* `<ENTER>`	*viewname* is the name of the view created in procedure 11.2.1.
3	Setup the toolkit environment.	`setenv PGSHOME` *`toolkithomedir`* `<ENTER>` `source $PGSHOME/bin/`*`sgiX`*`/pgs-dev-env.csh <ENTER>`	*toolkithomedir* is the home directory of the desired toolkit (see section 11.5.3). *X* in *sgiX* refers to the appropriate processor (see section 11.5.3 for details). Example: `source $PGSHOME/bin/sgi32/pgs-dev-env.csh <ENTER>`
4	Go into the directory containing the status message files.	`cd` *pathname* <ENTER>	*pathname* is the directory containing the SMF files. The user can be in any other directory while running `smfcompile`; however, full pathnames of the input files will have to be specified. Status Message Facility files have the extension .t. Example: MODIS_39501.t

Steps: (continued)

#	Purpose	Action	Comment

5	Run smfcompile on the SMF files.	`smfcompile -f` *`textfile`*`.t <ENTER>`
		or
		`smfcompile -f` *`textfile.t`* `-r -i <ENTER>`
			textfile is the filename of the status message text file delivered with the science software.
			Typically, the delivered status message textfiles should have filename extensions of *.t* and have the seed value used by the textfile as part of the filename (e.g. PGS_MODIS_39123.t, where 39123 is this message file's seed number).
			Only one textfile at a time can be processed; wildcards cannot be used.
			smfcompile may also be used with the -r and -i options. -r automatically copies the runtime files (PGS_XXXX) to the $PGSMSG directory. -i automatically copies the include files (PGS_INSTR_XXXX.h) to the $PGSINC directory.
			Note that if the source code being used is FORTRAN77, add the option `-f77` to produce FORTRAN include files. Likewise, `-ada` produces Ada include files.

6	Place the created include files and the created runtime ASCII files into the proper directories. This step is unnecessary if both the `-r` and `-i` options were used in the previous step.	`mv` *`includefilename`* `$PGSINC <ENTER>` `mv` *`runtimefilename`* `$PGSMSG <ENTER>`	*includefilename* is the name of the newly created include file (PGS_INSTR_XXXX.h) and *runtimefilename* is the name of the newly created runtime ASCII message file (PGS_XXXX). XXXX is the "seed" number that was in the corresponding .t file.
7	Repeat as necessary.	Repeat steps 5 and 6 as necessary for each of the SMF files associated with the science software.	Only one textfile at a time can be processed; wildcards cannot be used.

END OF PROCEDURE

11.6.3 Setting up a SDP Toolkit Environment

The purpose of the SDP Toolkit is to allow science software to be developed for ECS at independent Science Computing Facilities (SCFs). Version 5.1 of the SDP Toolkit was made available to the Instrument Teams in June 1996 so that science software built and linked to this SDP Toolkit could be tested at the SCFs. The science software delivered to the DAACs is expected to work with the SDP Toolkit installed at the DAACs.

There are several versions of the SDP Toolkit installed on the Science Processors (SGI Power Challenges) at the DAACs for Release A. Each version is independent in the sense that each contains a full directory structure similar to what was delivered to the Instrument Teams in SDP Toolkit Version 5.1 [Is this true still?}. The Toolkit versions at the DAACs differ according to:

Object Type

The operating system on the Science Processor SGI Power Challenges at Release A is IRIX 6.2 [Verify this!], a 64-bit operating system. To be backward compatible, the SGI operating system will allow both 64-bit and 32-bit objects to be built. In addition, there is a mode where objects can be built with compatibility for both IRIX 6.2 and the older IRIX 5.3 (used on the SGI's 32-bit machines) [Still true????]. Hence, there are three types of objects that can be built:

Object Type	cc Flag
New 32-bit	-n32
Old 32-bit	-32
New 64-bit	-64

where the cc Flag is the actual flag required on the command line to enable a particular mode with the SGI C compiler.

A version of the SDP Toolkit exists on the SGI Science Processor for each object type (see section 3 for more details).

Language Type

The SDP Toolkit uses different libraries depending upon whether FORTRAN 77 or Fortran 90 source code is being linked. If only C source code is to be linked, then either language version of the library will work.

Summary

Given that there are three object types (see **Object Type** above) and each has two language types (see **Language Type** above), six versions of the SDP Toolkit exist on the SGI Science Processors at the Release A DAACs. These versions are referred to collectively as the *DAAC Toolkit versions*. In addition, there are two versions of the SDP Toolkit that are functionally equivalent to the SDP Toolkit Version 5.1 delivered to the

Instrument Teams for development purposes. These are both old-32 bit (one for each language type) and are collectively referred to as the *SCF Toolkit versions*.

To summarize, the available SDP Toolkits are:

SDP Version	OS Compatibility	Language Type	Library Object Type	PGSHOME
SCF	IRIX 5.3	FORTRAN 77 or C	old 32-bit mode	/RelA_IT/scf_toolkit_f77/TOOLKIT
SCF	IRIX 5.3	Fortran 90 or C	old 32-bit mode	/RelA_IT/scf_toolkit_f90/TOOLKIT
DAAC	IRIX 5.3 & 6.1	FORTRAN 77 or C	old 32-bit mode	/RelA_IT/daac_toolkit_f77/TOOLKIT
DAAC	IRIX 5.3 & 6.1	Fortran 90 or C	old 32-bit mode	/RelA_IT/daac_toolkit_f90/TOOLKIT
DAAC	IRIX 6.1	FORTRAN 77 or C	new 32-bit mode	/RelA_IT/daac_toolkit_f77/TOOLKIT
DAAC	IRIX 6.1	Fortran 90 or C	new 32-bit mode	/RelA_IT/daac_toolkit_f90/TOOLKIT
DAAC	IRIX 6.1	FORTRAN 77 or C	new 64-bit mode	/RelA_IT/daac_toolkit_f77/TOOLKIT
DAAC	IRIX 6.1	Fortran 90 or C	new 64-bit mode	/RelA_IT/daac_toolkit_f90/TOOLKIT

where:

SDP Version

indicates whether the SDP Toolkit is a DAAC version or an SCF version. SCF versions are equivalent to SDP Toolkit 5.1

OS Compatibility

indicates under which operating system the Toolkit was compiled

Language Type

indicates for which languages the Toolkit is to be used

Library Object Type

indicates in which mode the Toolkit library was compiled

PGSHOME

indicates the location of the SDP Toolkit home directory appropriate for that version (Note: the assumption here is that the directory structure is the same on all Release A SGI Power Challenges).

Which DAAC Toolkit to Use

The choice of which version of the DAAC Toolkit to use mainly depends on which version is required by the science software. Most science code will likely require one of the 32-bit versions.

If science software is written in C only, then the language type does not matter and either language version of the DAAC Toolkit may be used. If, however, FORTRAN 77 code is being used (with or without C), then the FORTRAN 77 language version of the DAAC Toolkit must be

used. Conversely, if Fortran 90 code is being used (again, with or without C), the Fortran 90 language version of the DAAC Toolkit must be used.

If both FORTRAN 77 and Fortran 90 are being used, the procedure becomes more complex. Under such circumstances, refer to the *SDP Toolkit Users Guide for the ECS Project*.

Note that SSI&T procedures will likely include testing the delivered science software with the SCF Toolkit before testing with the DAAC Toolkit. For SCF Toolkit testing, there are only two versions, one for each language type.

Setting Up the SDP Toolkit Environment

The section describes how to set up a SDP Toolkit environment. There are eight environments available, one for each version of the SDP Toolkit (discussed _____). Setting up a SDP Toolkit environment means defining a number of environment variables to point to various locations within the particular SDP Toolkit hierarchy. By using these environment variables, build scripts or make files can be written to be independent of a particular SDP Toolkit version.

PGSHOME

The environment variable PGSHOME must be set on the Science Processor (SGI) to the home directory of the appropriate SDP Toolkit home directory. It is recommended that this variable *not* be set in any login startup script. The reason is that the user may be using several SDP Toolkit versions during routine SSI&T. Example:

C Shell:

```
setenv PGSHOME sdp_home_dir
```

Korn Shell:

```
PGSHOME=sdp_home_dir; export PGSHOME
```

where *sdp_home_dir* is the home directory of the particular SDP Toolkit version.

Other Environment Variables

The SDP Toolkit requires many other environment variables to be set. These can be set as:

C Shell:

```
source $PGSHOME/bin/sgiX/pgs-dev-env.csh
```

Korn Shell:

```
. $PGSHOME/bin/sgiX/pgs-dev-env.ksh
```

where *sgiX* should be replaced with one of:

sgi for old 32-bit versions of the SDP Toolkit

sgi32 for new 32-bit versions of the SDP Toolkit

sgi64 for new 64-bit versions of the SDP Toolkit

Example

To set the SDP Toolkit environment on the Goddard DAAC Science Processor for the new 32-bit version of the DAAC Toolkit where the source code is in both C and FORTRAN 77, the following sequences of commands can be issued:

C Shell:

```
setenv PGSHOME /vola1/RelA/daac_toolkit_f77/TOOLKIT
source $PGSHOME/bin/sgi32/pgs-dev-env.csh
```

Korn Shell:

```
PGSHOME=/vola1/RelA/daac_toolkit_f77/TOOLKIT
. $PGSHOME/bin/sgi32/pgs-dev-env.ksh
```

END OF PROCEDURE

11.6.4 Compiling a PGE and Linking With SCF Version of SDP Toolkit

Purpose: To produce a PGE by compiling delivered science software and linking it with the SCF version of the SDP Toolkit.

Tools/Materials/Assumptions: The science software delivery has been unpacked (e.g. uncompressed and/or untarred) and placed into the software build area. All files necessary to build the science software are available, accessible, and have the proper permissions set. This includes any install scripts, build scripts, make files, and instructions for using them. The build process will be done on the SGI Power Challenge in a UNIX shell (that is, outside of CASEvision or other COTS environments).

References:

System Description Document and Operations Manual from Instrument Team

Platform: SGI Power Challenge

Steps: The compiling and linking (building) of science software will vary according to the particular delivery. The instructions supplied with the delivery should be the primary source of information. What follows below are some "typical" steps that may or may not be applicable to a particular situation.

#	Purpose	Action	Comment
1	Read all instructional information supplied with the delivery. Such material should be the primary source of information on how to build science software.	Use any appropriate viewer or editor desired to read electronic documents. Read the Systems Description Document and the Operations Manual; both should be part of the delivery.	ASCII (text) files may be viewed with `more` or through `vi`. PostScript documents may be viewed through `ghostview` and PDF documents may be viewed through Acrobat Reader (both accessible via the SSIT Manager). Documents in Microsoft Word and related formats may be viewed through Microsoft Word (available via the SSIT Manager).
2	Start a new shell on the SGI Power Challenge science processor within which to build the software.	Either: From the SSIT Manager GUI, choose the Tools menu. Then choose xterm. Then telnet into the SGI Power Challenge. or In any currently available xterm window, spawn a new xterm session with: `xterm & <ENTER>` Then telnet into the SGI Power Challenge.	Starting a new shell avoids potential problems associated with redefinitions of environment variables.
3	Setup toolkit environment.	`setenv PGSHOME` *`toolkithomedir`* `<ENTER>` `source $PGSHOME/bin/`*`sgiX`*`/pgs-dev-env.csh <ENTER>`	*toolkithomedir* is the home directory of the desired toolkit (see section 11.6.3). *X* in *sgiX* refers to the appropriate processor (see section 11.6.3 for details). Example: `source $PGSHOME/bin/sgi32/pgs-dev-env.csh <ENTER>`

4	If necessary, set the ClearCase view and check out the make file.	`cleartool setview` *`viewname`* `<ENTER>`
		`cd` *`path`* `<ENTER>`
		`cleartool checkout -nc` *`makefile`* `<ENTER>`
			viewname is the name of the view created in procedure 11.3.1.
			path is the name of the directory in the VOB where the make file was checked in.
			makefile is the name of the make file to be used with this PGE.
			-nc stands for "no comment", which means that ClearCase will not prompt for comments.

Steps: (continued)

#	Purpose	Action	Comment

5	Examine and alter (if necessary) any supplied make files.	Invoke favorite text editor (`vi`, `emacs`, `xedit`, etc.)
		Check that compiler, compiler flag settings and other environment variable settings are appropriate.
		Add the following components to the library line:
		`$(ADD_LFLAGS)`
		`$(ADD_LIBS)`
			The delivery may include several versions of make files appropriate to different platforms. Refer to supplied documentation.
			The delivery may include install scripts which set environment variables.
			Try to make use of environment variables pre-set used by the shell (e.g. F77, C, CFLAGS) rather than setting them within the make/build files.

6	Compile any Status Message Facility (SMF) files.	See procedure 11.5.2.	
7	Verify that the directory structure matches the directories expected by the build script.		Deliveries may come with install scripts which place files into various directories according to some structure.
8	If an executable or object files for this PGE are already checked into ClearCase, check it or them out.	`cleartool checkout -nc` *`filename`* `<ENTER>`	*filename* is the name of the executable or object file to be checked out. It is *not* recommended that executables and object files be checked into ClearCase; they take up too much room, can easily be rebuilt, and make compiling and running unnecessarily complicated when they are checked into ClearCase. -nc stands for "no comment", which means that ClearCase will not prompt for comments.
9	Once installation scripts, make files and/or build scripts are setup properly, perform the build according to the instructions delivered with the PGE.		Science software deliveries may come with a single, top-level script to do the entire build or the build process could involve a series of steps, each of which should be described fully in the delivered documentation.
10	If the make file had been checked out of ClearCase, then check the modified version back into ClearCase.	`cleartool checkin` *`makefile`* `-nc <ENTER>`	*makefile* is the name of the make file to be used with this PGE. -nc stands for "no comment", which means that ClearCase will not prompt for comments.

END OF PROCEDURE≠

11.6.5 Compiling a PGE and Linking with DAAC Version of SDP Toolkit

Purpose: To produce a PGE by compiling delivered science software and linking it with the DAAC version of the SDP Toolkit.

Tools/Materials/Assumptions: The science software was previously built successfully using the SCF version of the SDP Toolkit.

If this is not the case, then the science software delivery has been unpacked (e.g. uncompressed and/or untarred) and placed into the software build area. All files necessary to build the science software are available, accessible, and have the proper permissions set. This includes any install scripts, build scripts, make files, and instructions for using them. The build process will be done on the SGI Power Challenge in a UNIX shell (that is, outside of CASEvision or other COTS environments).

References:

System Description Document and Operations Manual from Instrument Team

Platform: SGI Power Challenge

Steps: The compiling and linking (building) of science software will vary according to the particular delivery. The instructions supplied with the delivery should be the primary source of information. What follows below are some "typical" steps that may or may not be applicable to a particular situation.

Most of the following steps will be unnecessary if the science software had been previously built successfully with the SCF version of the SDP Toolkit in procedure 11.6.4.

#	Purpose	Action	Comment
1	Read all instructional information supplied with the delivery. Such material should be the primary source of information on how to build science software. This step may not be necessary if the software had been previously built successfully with the SCF version of the SDP Toolkit.	Use any appropriate viewer or editor desired to read electronic documents. Read the Systems Description Document and the Operations Manual; both should be part of the delivery.	ASCII (text) files may be viewed with `more` or through `vi`. PostScript documents may be viewed through `ghostview` and PDF documents may be viewed through Acrobat Reader (both accessible via the SSIT Manager). Documents in Microsoft Word and related formats may be viewed through Microsoft Word (available via the SSIT Manager).
2	Start a new shell on the SGI Power Challenge science processor within which to build the software.	Either: From the SSIT Manager GUI, choose the Tools menu. Then choose Xterm. Then telnet into the SGI Power Challenge. or In any currently available xterm window, spawn a new xterm session with: `xterm & <ENTER>` Then telnet into the SGI Power Challenge.	Starting a new shell avoids potential problems associated with redefinitions of environment variables.
3	Setup the toolkit environment.	`setenv PGSHOME` *`toolkithomedir`* `<ENTER>` `source $PGSHOME/bin/`*`sgiX`*`/pgs-dev-env.csh <ENTER>`	*toolkithomedir* is the home directory of the desired toolkit (see section 11.6.3). *X* in *sgiX* refers to the appropriate processor (see section 11.5.3 for details). Example: `source $PGSHOME/bin/sgi32/pgs-dev-env.csh <ENTER>`

4	If ClearCase is being used for CM, set the ClearCase view and check out the make file.
	This step should be unnecessary if the software had been previously built successfully with the SCF version of the SDP Toolkit because the makefile will not have to be modified.
		`cleartool setview` *`viewname`* `<ENTER>`
		`cd` *`path`* `<ENTER>`
		`cleartool checkout` *`makefile`* `-nc <ENTER>`
			viewname is the name of the view created in procedure 11.3.1.
			path is the name of the directory in the VOB where the make file was checked in.
			makefile is the name of the make file to be used with this PGE.
			-nc stands for "no comment", which means that ClearCase will not prompt for comments.

Steps: (continued)

#	Purpose	Action	Comment

5	Examine and alter (if necessary) any supplied make files.
	This step should be unnecessary if the software had been previously built successfully with the SCF version of the SDP Toolkit.
		Invoke favorite text editor (`vi`, `emacs`, `xedit`, etc.)
		Check that compiler, compiler flag settings and other environment variable settings are appropriate. Try to use the same settings as the SDP Toolkit. Assuming step 2 was done, use the command `env` to list current settings of the shell's environment variables.
		Add the following components to the library line:
		`$(ADD_LFLAGS)`
		`$(ADD_LIBS)`
			The delivery may include several versions of make files appropriate to different platforms. Refer to supplied documentation.
			The delivery may include install scripts which set environment variables.
			Try to make use of environment variables used by the shell (e.g. F77, C, CFLAGS) rather than setting them within the make/build files.

6	Compile any Status Message Facility (SMF) files.	See procedure11.5.2.	Note that while this step may have been performed in compiling the software with the SCF Toolkit, part of it still needs to be performed for the DAAC Toolkit; specifically, steps 6 and 7 of procedure 11.6.2 must be repeated AFTER the DAAC Toolkit environment has been set up so that the message and header files are placed in the DAAC version of the Toolkit.
7	Verify that the directory structure is as intended. This step will be unnecessary if the software had been previously built successfully with the SCF version of the SDP Toolkit.		Deliveries may come with install scripts which place files into various directories according to some structure.
8	Once installation scripts, make files and/or build scripts are setup properly, perform the build according to instructions, preferably using clearmake.		Science software deliveries may come with a single, top-level script to do the entire build or the build process could involve a series of steps, each of which should be described fully in the delivered documentation.
9	If ClearCase is being used for CM and the make file was checked out in step 4, check the make file back into ClearCase.	`cleartool checkin -nc` *`makefile`* `<ENTER>`	*makefile* is the name of the make file to be used with this PGE. -nc stands for "no comment", which means that ClearCase will not prompt for comments. If the make file was checked out but not modified, see procedure 11.3.6 for instructions on unchecking out the file.

END OF PROCEDURE

11.7 Running a PGE in a Simulated SCF Environment at the DAAC≠

11.7.1 Setting Up the Environment for Running the PGE≠

11.7.2 Running and Profiling the PGE≠

11.8 Updating the PDPS Database and Data Server≠

11.8.1 Updating the Science Metadata in the PDPS Database≠

11.8.2 Updating the ESDT Metadata in the PDPS Database≠

11.8.3 Updating the Operational Metadata in the PDPS Database≠

11.8.4 Inserting Test Data Files Into the Data Server≠

11.8.5 Inserting Static Files into the Data Server≠

11.8.6 Inserting the Science Software Executable Package (SSEP) into the Data≠ Server≠

11.9 PGE Planning and Processing≠

11.9.1 Using the Planning Workbench≠

11.9.2 Using the Production Request Editor≠

11.9.3 Registering a Subscription for Test Output Files≠

11.9.4 Monitoring Production≠

11.9.5 Acquiring the Test Output Files from the Data Server≠

11.10 File Comparison≠

11.10.1 Using the HDF File Comparison Tool GUI≠

Purpose: To compare two HDF files

Tools/Materials/Assumptions: Two HDF files exist with similar structures, or at least having a data set in common. The SSIT Manager is running on a Sun workstation. If the files are in ClearCase, a ClearCase view was set before the SSIT Manager was started.

References:

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Bring up the HDF file comparison tool.	From the SSIT Manager GUI, select the Tools menu. Then select Product Examination, then File Comparison, and then HDF.	This will start the HDF File Comparison GUI.

2	Load the first HDF file.	Click on the File #1 button.	Start with the path name. Enter it segment by segment. For example:
			`/home <ENTER>`
			`/toma <ENTER>`
			`/ssit <ENTER>`
			Set the filter to `*.hdf`.
			Highlight a filename and click OK.

3	Load the second file.	Select file 2 using screen.	See step 2.
4	Pick the data set to be compared within the file, and view the differences if any.	Look in the center window at list of common data sets. Click on one.	The window below shows the differences.
5	Optionally, view the file attributes.	Click on a button under the filename.	The attributes or metadata are displayed.
6	Continue, if desired, on additional data sets.	Click on more data sets, one by one.	
7	Obtain a report.	Click on the REPORT button.	The report indicates the data items that differ, numbered by "C" style row and column. This is saved to a file - rename it or it will be overwritten.
8	WARNING	If the Report Window is closed, the session may end.	Sometimes trying a new report or closing produces core dumps.
9	Make a plot of the differences.	Click on a member of the row of little buttons just above the screen of differences. Then click one of the PLOT buttons.	An IDL plot of the differences will appear - data set 1 versus 2 or set 1 minus set 2 versus index.
10	Change the tolerances for comparison.	Click on Tolerances.	Follow the instructions to set the relative or absolute tolerance.

END OF PROCEDURE

11.10.2 Using theHDF File Comparison Tool hdiff

11.10.3 Using the ASCII File Comparison Tool

Purpose: To compare two ASCII files.

Tools/Materials/Assumptions: Two ASCII files exist. The SSIT Manager should be running. If the files are in ClearCase, a view was set before the SSIT Manager was started.

Note: when this is attempted on files with a long line, the tool may seem to show only one file - the other is far off to the right. *Observe the bottom scroll bar carefully.*

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Bring up the ASCII File Comparison tool.	From the SSIT Manager GUI, select the Tools menu. Then select Product Examination, then File Comparison, and then ASCII.	This will start the ASCII Comparison GUI.
2	Enter the first file name.	*path/filename* <ENTER>	*path* is the complete directory path where the file is located. *filename* is the name of the first file to be compared.
3	Enter the second file name.	*path/filename* <ENTER>	*path* is the complete directory path where the file is located. *filename* is the name of the second file to be compared.
4	View any differences in the files.	Visually compare the left and right windows.	Use the scroll bar to see any data that is off the window on the right. If no window appears, files are identical and a new prompt will appear in the window.
5	Repeat steps 2 through 4 for any more file pairs. Quit when done.	Q	Typing Q will quit the comparison tool.

END OF PROCEDURE

11.10.4 Using the Binary File Comparison Tool

Purpose: To compare the contents of two binary files.

Tools/Materials/Assumptions: Two binary files exist. The SSIT Manager should be running. If the files are in ClearCase, a ClearCase view was set before the SSIT Manager was started. The file format is known.

This procedure explains how to modify sample file comparison code. If a binary file comparison utility was delivered with the science software, this procedure will not be necessary.

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Bring up the Binary File Comparison tool.	From the SSIT Manager GUI, select the Tools menu. Then select Product Examination, then File Comparison, and then Binary.	This will start the Binary File Comparison GUI.
2	Select a language from among C, FORTRAN, or IDL.	Leave the default (C) or click on FORTRAN or IDL.	This is the language to be used in the file comparison tool that is being custom-built in this procedure.
3	To view the sample code: select sample code to view.	Click on a button under "Compare Function", "Driver", or "Utilities" (in blue). The buttons are marked.	Note that the three blue headings are alternate choices, and under them are buttons for further choice. Viewing and copying cannot be done at the same time.
4.0	To copy the sample code to disk: select sample code to write to disk both driver and function.	Click on Copy.	A window prompts for a "file identifier". This could be the name of the instrument. Viewing and copying cannot be done at the same time.
4.1	Identify the code by supplying a string to embed in the new file name.	Enter identifier.	The string will be embedded in filenames (function and driver) and inside the code.
4.2	Customize the code to work with the data structure of these files.	Edit code to conform to this PGE's binary file structure using any editor.	The sample code is just an example, because data files are intelligible only when the code "understands" their structure.

END OF PROCEDURE

11.11 Data Visualization

11.11.1 Viewing Product-Created Metadata Using the EOSView Tool

Purpose: To allow a user to check metadata of a data product created by running a PGE.

Tools/Materials/Assumptions: The science software being used to generate a data product has been ingested, built and successfully run in the Release A environment. The product created has been written out as a HDF File (**not HDF-EOS at this point**) and contains metadata. The SSIT Manager Tool can be used to used to start up the ECS data browser EOSView.

References:

SSIT Help Pages URL:

EOSView Documentation - On-Line Help from the Main EOSView Dialog Window.

HDF-EOS Primer for Version 1 EOSDIS (White Paper 175-WP-001-001)

NCSA HDF Reference Manual on HDF Version 3.3

Platform: Sun

Steps:

#	Purpose	Action	Comment
1	Start up EOSView to view the Product Metadata.	From the SSIT Manager GUI, select the Tools menu. Then select Production Examination, then EOSView.	This will start the EOSView GUI. Alternatively, EOSView can be started from the command line. To do this, within an xterm: `set path = ($path $DPATMGR_HOME/bin/sun5/EOSView) <ENTER>` `setenv UIDPATH $DPATMGR_HOME/data/eosview.uid <ENTER>` where DPATMGR_HOME is set to the path of the SSIT Manager (see section 11.3.1 for details). `EOSView & <ENTER>`
2	Open a HDF product file from which metadata is to be viewed.	From the main EOSView window, select the File Open button from the main menu bar. A File Selection Dialog window should open and the user should select the appropriate directory and file to open.	
3	Once the desired product file has been opened, the specific types of HDF objects in the file will be listed in the Contents window.	By viewing the HDF objects within the product file, a user can select a particular HDF Object (Vgroup, SDS, etc.), to get to the metadata (in HDF terminology, metadata is referred to as *attribute data*).	
4	Choose the HDF object for which you wish to view the metadata.	Double click on the HDF object. The structure of the object will appear in a dialog window with buttons on the bottom portion of the window to view the data of the object itself.	
5	Display the attribute (i.e., metadata) of this particular HDF object.	In the window displaying the structure of the HDF object, select the Table button to display the table data of the object.	
6	View the attribute (metadata) of a particular HDF object.	This window will display the table data found in the HDF object being examined. From the main menu bar, select the File-Attribute option of the main menu bar.	Any metadata associated with the object will be displayed in another text window.

END OF PROCEDURE

11.11.2 Viewing Product Data with the EOSView Tool≠

11.11.2.1 Viewing HDF Image Files≠

11.11.2.2 Viewing HDF Vdata Files and HDF-EOS Files≠

11.11.2.3 Outputting HDF to Other Formats≠

11.11.3 Viewing Product Data with the IDL Tools≠

11.11.3.1 Managing Input and Output≠

11.11.3.2 Creating and Saving Plot Displays≠

11.11.3.3 Creating and Saving Image Displays≠

11.11.3.4 Image Processing Fundamentals≠

11.11.3.5 Raster Mapping Fundamentals≠

11.11.3.6 Volumetric Data Analysis≠

11.12 Post-Testing Activities≠

11.12.1 Validating Inventory/Granule Metadata Updates to the Data Server≠ Database≠

11.12.2 promoting the PGE from Test to Production≠

11.12.3 Reporting Science Software Problems≠

11.12.4 Reporting ECS Software Problems≠

11.13 Science Software Archive Package (SSAP)≠

11.13.1 Assembling Components of the SSAP≠

11.13.2 Creating and Inserting the SSAP into the Data Server≠

11.14 Troubleshooting and General Investigation≠

11.14.1 Examining PGE-Produced Log Files≠

Purpose: To check for error, warning and status messages written by the SDP Toolkit or the science software using the SDP Toolkit Status Message Facility (SMF).

Tools/Materials/Assumptions: The Data Processing Request ID (DPR ID) for the PGE execution is required for this procedure. The SDP Toolkit writes its log files to the directories specified in the instantiation of the Process Control File (PCF). These log file pathnames may be known to operations staff who are running the SSIT procedures. If not, the location of the log files may be obtained either from the instantiated PCF if its full path is known or they may be obtained from the PDPS Database, knowing the DPR ID.

The instantiated PCF contains a line for each type of log file. Field 2 contains the log file name. The location of the log files may be specified in field 3. If field 3 is blank, the log files should go to the default $PGSHOME/runtime. The PC file identifies each of the following logs:

LogStatus Status log file which captures all error and status information concerning a program.

LogUser User log file which captures a subset of messages of level or type "_U_" or "_N_" which are of particular interest to the user. This file may also contain notices, generally for data availability.

LogReport Report log file which captures arbitrary message strings sent by the PGE software. The messages are unrelated to the toolkit functions. The report log can contain very important messages for scientists and developers.

The assumptions for this procedure are that all required environmental variables have been set and that the operations/test account has "**read**" access to the PDPS Database. The .cshrc file should have the following entries (see section 3.1.1 for details):

```
setenv SYBASE/vendor/sybase

setenv SYBROOT $SYBASE/sybooks

setenv EBTRC $SYBROOT/sun5m/.ebtrc

setenv DSQUERY computer-server

( where computer-server is system dependent)

set path = ($path $SYBASE $SYBASE/bin $SYBROOT/sun5m/bin)
```

References: SDP Toolkit Primer for the ECS Project, 194-815-SI4-001

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	If the full pathnames of the log files are known, proceed to step 11. If not and the full path of the instantiated PCF is known, proceed to step 5. If the path of the PCF is not known, proceed to step 2.		
2	Examine the instantiated PCF for the location of the log files.	`cd` *`pathname`* `<ENTER>` `xedit` *`pcfile`* `<ENTER>`	*pathname* is the pathname of the instantiated PCF. *pcfile* is the name of the instantiated PCF. Editors should include vi, emacs, and xedit.
3	Either locate the SUPPORT OUTPUT FILES or the logical numbers associated with each log file.	Search for SUPPORT OUTPUT FILES or 10100, 10101, and 10102	Logical numbers refer to the log files: 10100 LogStatus 10101 LogReport 10102 LogUser
4	When finished examining the PCF, quit out of the editor. If the full paths of the log files have been obtained from the PCF, proceed to step 11. If not, proceed to step 5.	Hit the "Quit" pull down button for xedit.	

5	If not already logged into the PDPS Database server (DPS Sun), log into it.	`xterm &` `<ENTER>`
		`telnet` *`IP-address`* `<ENTER>`
		Enter login name and password.	

6	The SQL procedure to be run under the database displays the values in all tables related to the DPR ID. The output can be large. If a hardcopy is desired, make a script file of the screen output and re-source the .cshrc file. If a screen output is desired, skip this step.	`script` *`reportname`* `<ENTER>` `source .cshrc <ENTER>`	*reportname* is a user-defined name for the file. The script command starts a shell. The database environmental variables described above may need to be set again.
7	Log into the PDPS Database.	`isql -U`*`log-name`* `<ENTER>` Enter password.	
8	Run the SQL procedure to display the PDPS database information.	`sp_view_dpr` *`dprid`* `<ENTER>`	*dprid* is the id number of the dpr to which the log files belong.
9		`go <ENTER>`	Execute the SQL command.

Steps: (continued)

#	Purpose	Action	Comment

10	If a script report file has been made, quit the database, exit the shell for the script, examine the output under an editor and then quit the editor.
	If a screen output only is made, scroll to examine the output and then quit the database.
		If a script report is made, do the following:
		`quit`
		`exit`
		`xedit` *`reportname`* `<ENTER>`
		For either report or screen output:
		Obtain full pathnames and log file names of the log files as listed under the SUPPORT OUTPUT FILES heading.
		`quit`
			The database must be exited and the report file closed before the report file can be examined for the log file names and paths. *reportname* is the user-defined name of the report file.
			Alternatively, any editor may be used or the *more* command.
			The pathnames and log file names are obtained by visual examination of the report file or the screen output.
			Quit the editor for the report file or the database for the screen output.

11	Log into the SGI Power Challenge.	From the SSIT manager GUI, choose Tools menu. Then choose xterm. Telnet to the SGI Power Challenge.	The science software will be executed on the SGI Power Challenge, thus, the log files will be accessed from there.
12	Go to the status log directory to examine LogStatus file	`cd` *`pathname`* `<ENTER>`	*pathname* is the pathname of the LogStatus file (either known to operations/test staff or as obtained in step 3 or 10).
13	Examine the LogStatus file for error, warning, and status messages.	`xedit` *`logstatus`* `<ENTER>`	*logstatus* is the name of the LogStatus file (either known to operations/test staff or as obtained from step 3 or 10). Alternatively, any editor may be used or the *more* command.
14	If finished examining the file, quit out of the editor.	Hit the "Quit" pull down button (for xedit).	
15	Go to the user log directory to examine the LogUser file.	`cd` *`pathname`* `<ENTER>`	*pathname* is the pathname of the LogUser file (either known to the operations/test staff or as obtained from step 3 or 10).
16	Examine the LogUser file for generated messages and notices.	`xedit` *`loguser`* `<ENTER>`	*loguser* is the name of the LogUser file (either known to operations/test staff or as obtained from step 3 or 10). Alternatively, any editor may be used or the *more* command.
17	If finished examining the file, quit out of the editor.	Hit the "Quit" pull down button (for xedit)	

Steps: (continued)

#	Purpose	Action	Comment
18	Go to the report log directory to examine the LogReport file.	`cd` *`pathname`* `<ENTER>`	*pathname* is the pathname of the LogReport file (either known to operations/test staff or as obtained in step 3 or 10).
19	Examine the Log Report file for arbitrary messages sent by the PGE software.	`xedit` *`logreport`* `<ENTER>`	*logreport* is the name of the LogReport file (either known to operations/test staff or as obtained from step 3 or 10). Alternatively, any editor may be used or the *more* command.
20	If finished examining the file, quit out of the editor.	Hit the "Quit" pull down button (for xedit).	

11.14.2 Examining the MSS Log File

Purpose: To examine the Management Subsystem log files.

Tools/Materials/Assumptions: MSS Server, HP Openview

References: DID609, Section 6.2.1.3.5

Platform: Sun or SGI Power Challenge

Steps:

#	Purpose	Action	Comment
1	If needed (e.g., Sun), set the workstation for use as an xserver by MSS workstation.	`xhost +` *`MSSservername`* `<ENTER>`	*MSSservername* is the name or IP address of the MSS server.
2	Connect to the MSS Server.	Telnet to the local MSS server.	
3	Set the environmental variable DISPLAY to the name or address of the display terminal.	`setenv DISPLAY` *`xtermname`*`:0.0 <ENTER>`	*xtermname* is the name or IP address of the display terminal.
4	Initiate an HP Openview session.	`/usr/OV/bin/ovw & <ENTER>`	
5	Set the criteria for those events that are to be displayed.	Click on the check box for the events types to be examined. Select “Set Filters” under the View menu.	Science software events are “Application Alert Events”.

Steps: (continued)

#	Purpose	Action	Comment
6	Set the earliest events to be displayed.	Click on the Check Box "Received after Time/Date". Select either “Coarse Granularity” or “Fine Granularity” from the pop-up menu. Adjust the slide to select the earliest Date/Time.	
7	Set the latest events to be displayed.	Click on the Check Box “Received before Time/Date”. Select either “Coarse Granularity” or “Fine Granularity” from the pop-up menu. Adjust the slide to select the latest Date/Time.	
8	Set the message string to be matched.	If further filtering by message string is desired, click on the check box “Match Message String”. Enter any string to be matched into the text entry field.	Performed only if “Match Message String” is checked.
9	View the results of the selection criteria.	Click on Apply.	
10	Modify selection criteria, if.	Add or change time span or match string, the click again on Apply.	
11	Complete the filtering process.	Click on OK.	Clicking on Cancel returns to the full list of events.
12	Exit HP Openview.	Select “EXIT” from the “File” Menu.	
13	Logout from the MSS server.	`logout <ENTER>`	

END OF PROCEDURE

11.14.3 Using PDPS Prototype-Related Scripts and Examining Message Files

Purpose: To point users to files which are created during the execution of a DPR job and which may hold information useful in tracing processing problems.

Tools/Materials/Assumptions: Some of these files are written by default to directory paths that can only be accessed on either the SGI processor machine or one of the Sun workstations. More detailed descriptions of these files and the conditions under which they are generated will be supplied in future Green Book versions.

Platform: Sun or SGI

Steps:

(1) /tmp/JILxxxxxxxxx (only accessible on Sun)

Description:

JILxxxxxxxxx is the Job Information Language (JIL) script that defines the DPR job to AutoSys and which must be submitted to the AutoSys Database before a DPR job can be run. The name of the file created is system-generated and begins with the characters 'JIL' followed by nine characters (e.g. JILAAAa0066c). In Ir1 PDPS release 1.03, the JIL script for a given DPR job is written to a file under the /tmp directory and not to a TempJilScript file, as was done in previous Ir1 PDPS releases.

Sample file content:

insert_job: 5251_823122483_1

job_type: command

command: /data/Ir1/AI_T/bin/sgi/DpAtExecutionMain 5251_823122483_1

machine: spr1sgigsfc

std_out_file: /home/cboettch/mockpge_msfc/out/dpat_std.out

std_err_file: /home/cboettch/mockpge_msfc/out/dpat_std.err

profile: /data/Ir1/AI_T/bin/sgi/DpAtRunProfile.sh

(2) */user/defined/directory/filename1* for DpAtExecutionMain stdout (Default directory path only accessible on SGI)

(3) */user/defined/directory/filename2* for DpAtExecutionMain stderr (Default directory path only accessible on SGI)

filename1 (e.g. DpAtExecutionMain.out) and *filename2* (e.g. DpAtExecutionMain.err) contain the PDPS Prototype-level error and status messages generated during the execution of a DPR job, as opposed to PGE-level error and status message written to the LogStatus, LogReport and LogUser files described in section 11.8.1. These two files can be written to a directory on a SGI disk volume when a DPR job is executed. DpAtExecutionMain.out contains stdout messages resulting from the PGE itself; these may be generated from a user script which executes the PGE and may contain different information from that in the Log* files. DpAtExecutionMain.err contains stderr messages resulting from the execution of the PDPS prototype (i.e. the executable DpAtExecutionMain.) In Ir1 PDPS build release 1.03, the information represented by these two

files are written by default to /dev/null, but the user can ensure that the files are generated by setting the environment variables DPAT_STD_ERR and DPAT_STD_OUT. The location and names of these files can be user-defined by setting the environment variables DPAT_STD_ERR and DPAT_STD_OUT. Setting these variables locally ensures that only a user's own DPR job messages are written to these files. DPAT_STD_ERR and DPAT_STD_OUT are not set automatically and the files will not be created automatically. If these files are desired, the user must explicitly set DPAT_STD_ERR and DPAT_STD_OUT *prior* to bringing up the SSIT Manager and executing the PDPS prototype software and in a manner such as:

setenv DPAT_STD_ERR */user/defined/directory/filename1*

setenv DPAT_STD_OUT */user/defined/directory/filename2*

Sample file contents for DpAtExecutionMain.err:

UX:sh (/bin/sh): ERROR: /RelA_IT/AI_T/bin/sgi/DpAtExecutionMain: Cannot execute

SCIENCE_CODE_ERROR...PGE's exit status shows PGE abnormal end or PGS_PC_Shell.sh not found.:PGS_PC_SHELL exit status = 15

Errid = DpAt|0.1|25232|IRIX64|6.1|spr1sgigsfc|192.107.190.90|

SYSTEM_MESSAGE...Process Information: Path to Resource Profile Output is invalid, Resource Profile Output File not written.

Error: PGE file must exist and have size greater than zero!

Errid = DpAt|0.1|18971|IRIX64|6.1|spr1sgigsfc|192.107.190.90|

SCIENCE_CODE_ERROR...PGE's exit status shows PGE abnormal end or PGS_PC_Shell.sh not found.:PGS_PC_SHELL exit status = 246

Errid = DpAt|0.1|2711|IRIX64|6.1|spr1sgigsfc|192.107.190.90|

SYSTEM_ERROR...Process Information: Path to Process Control File (PCF) is invalid.

Errid = DpAt|0.1|19181|IRIX64|6.1|spr1sgigsfc|192.107.190.90|

SCIENCE_CODE_ERROR...PGE's exit status shows PGE abnormal end or PGS_PC_Shell.sh not found.:PGS_PC_SHELL exit status = 15

UX:csh: ERROR: test.exe - Command not found

(4) /*user*/*defined*/*directory*/eventLogger.log

Description:

This file is the file indicated by the following science software PCF entry in the SUPPORT OUTPUT FILES section:

10113|eventlogger.log|||||1

Because this file is specified in the science software PCF, the user can define both the directory path and the file name.

Sample file contents:

7 7 3 4 1 | 2 | 0 | 0 1 / 1 8 / 9 6 1 3 : 4 0 : 0 8 | S D P T o o l k i t | D A A C Rel|21395|IRIX64|6.1|dps1sgiedf|192.150.28.111|21395|Action String: PGE Processing will NOT be performed.

Return status: PGSPC_C_PGSINFO_PGE_STOP

Message: Toolkit Initialization Procedure has failed!

|

7 5 2 9 4 | 4 | 0 | 0 1 / 1 8 / 9 6 1 3 : 4 0 : 0 8 | S D P T o o l k i t | D A A C Rel|21397|IRIX64|6.1|dps1sgiedf|192.150.28.111|21397|Action String: Toolkit Termination Procedure is underway.

Return status: PGSPC_C_PGSINFO_TERM_START

Message: Initialization or PGE Processing has completed.

|

7 5 2 9 2 | 4 | 0 | 0 1 / 1 8 / 9 6 1 4 : 0 3 : 5 8 | S D P T o o l k i t | D A A C Rel|21865|IRIX64|6.1|dps1sgiedf|192.150.28.111|21865|Action String: PGE Processing starting up.

Return status: PGSPC_C_PGSINFO_PGE_START

Message: Toolkit Initialization Procedure has succeeded!

|

7 5 2 9 4 | 4 | 0 | 0 1 / 1 8 / 9 6 1 4 : 1 2 : 4 4 | S D P T o o l k i t | D A A C Rel|21940|IRIX64|6.1|dps1sgiedf|192.150.28.111|21940|Action String: Toolkit Termination Procedure is underway.

Return status: PGSPC_C_PGSINFO_TERM_START

Message: Initialization or PGE Processing has completed.

|

(5) /usr/local/hislog/pdps_event.log (Default directory path only accessible on each Sun and SGI)

Description:

This file is pointed to by the environment variable DPAT_EVENTLOG and is written to as DpAtExecutionMain executes.

Sample file contents:

100860012|3|1|01/02/96 09:35:43|DpAt|0.1|1411|SunOS|5.4|ait1sungsfc|192.107.190.93|1411|Can't find message #8196204

Check that $PGSMSG points to PGS_10xx files|

100862469|2|1|01/05/96 10:35:35|DpAtMgr|1.0|9352|SunOS|5.4|ait1sungsfc|192.107.190.93|9352|Cannot find program DpAtMgrBinDiff|

100862469|2|1|01/05/96 10:39:12|DpAtMgr|1.0|9352|SunOS|5.4|ait1sungsfc|192.107.190.93|9352|Cannot find program EOSView|

100860015|3|1|01/16/96 13:13:58|DpAt|0.1|2291|SunOS|5.4|ait1sungsfc|192.107.190.93|2291|SYSTEM_ERROR...Unable to get PGE Names from PGE Database|

18:01:15|DpAt|0.1|6533|SunOS|5.4|ait1sungsfc|192.107.190.93|6533|INPUT_ERROR...Unable to read from specified Process Control file|

100860078|3|1|02/01/96 12:13:28|DpAt|0.1|13689|SunOS|5.4|ait1sungsfc|192.107.190.93|13689|INPUT_ERROR...Unable to read from specified Process Control file|

END OF PROCEDURE

11.15 Miscellaneous

11.15.1 Setting Up the Release A Bulletin Board Service

11.15.2 Obtaining Information from the Bulletin Board Service

11.15.3 Posting Information to the Bulletin Board Service

12. Resource Planning

The Resource Planning process is the mechanism by which reservations for non-routine ground events are defined and controlled. Such events may include science data processing and distribution, testing, simulations, preventive maintenance or upgrades, or any other event that requires DAAC 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 10-day plans, and on a daily basis. However, ground events can be entered at any time. Resource plan reports can be published to the Document Data Server for review by parties outside the DAAC in the same way that production plans are.

The site M&O Resource Planner uses the Resource Planning Workbench utility within the Planning Subsystem to schedule non-routine events against ECS resources.

In addition to the Resource Planning Workbench utility, Resource Planning provides a Resource Definition tool to add resources or to modify the characteristics of the resources. Step-by-step procedures for using the Resource Planning Workbench utility are presented in Section 12.2. Step-by-step procedures for using the Resource Definition tool are presented in Section 12.3

12.1 Resource Planning Process

Resources which are defined and subsequently used in production planning, are strings (computers and storage devices); computers; and disks. Other generic hardware devices may also be defined and managed. The Resource Planning subsystem is initialized with the resource data from the Baseline Manager database. Resources may be added to or deleted from the resource planning list without affecting the Baseline Manager database. This is useful, for example, for identifying resources that will be available in the future.

The Resource Planner views requests for resource reservations to determine if the requests are valid. Requests include information, such as activity description, resource requirements, and time requested to use the resource (acceptable variances may be specified in comments field). The Resource Planner may decide to forward the request to a "subject-matter-expert," whose expertise is particularly relevant to the request, in order to validate a request. (The subject-matter-expert is referred to as the "sponsor" by the system; see Section 12.2.1, Step 3, below.)

Should the subject-matter-expert agree that the request to reserve the resource is valid, the Resource Planner will approve it along with all other requests that have been validated, and generate a Resource Plan. The Plan will be submitted to a review board (members to be determined) for the board's consideration. The review board will confirm that the reserved resources will not adversely impact scheduled events. Once approved by the review board, the Resource Planner will "commit" the Plan. The system will then automatically forward the Plan to Production Planning for scheduling.

Table 12.1-1, below, provides an Activity Checklist table, of Resource Planning activities. 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 12.1-1. Resource Planning - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner or DAAC Staff	Create a Resource Reservation Request	(P) 12.2.1	
2	Resource Planner	Edit a Resource Reservation Request	(P) 12.2.2	
3	Resource Planner or Sponsor	Validate a Resource Reservation Request	(P) 12.2.3	
4	Resource Planner	Approve a Resource Reservation Request	(P) 12.2.4	
5	Resource Planner	Activate a Resource Reservation Plan	(P) 12.2.5	
6	Resource Planner	Publish/Review Resource Plan	(P) 12.2.6 12.2.7	
7	Resource Planner	Delete a Resource Reservation Request	(P) 12.2.8	
8	Resource Planner	Modify Resource Planning List	(P) 12.3	

12.2 Resource Planning Workbench Utility

The workbench utility is invoked by **double-clicking** on the Resource Planning Workbench icon from the ECS desktop. The Resource Planning screen (graphical user interface) is accessed. This screen contains the Resource Reservation List, activity type, and a series of buttons that enable the following operations:

New... Create a resource reservation request. (Section 12.2.1)

Modify... Edit or review the details of an existing resource reservation request. (Section 12.2.2)

Delete Mark a resource reservation as deleted. (Section 12.2.8)

Approve Mark a resource reservation request as approved. (Section 12.2.4)

Commit Commit all approved resource reservations; at this point the ground events will be accessible by the production planning software. (Section 12.2.5)

Timeline Display a timeline oriented view of the resource plan. (Section 12.2.7)

Report Display a GUI for selecting reports and report options. (Section 12.2.6)

12.2.1 Create a Resource Reservation Request

Table 12.2-1 presents the steps required to create a Resource Reservation in a condensed manner. 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 Resource Reservation GUI by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, **click** on the **New...** button to access the Resource Request Edit screen.

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.

- **a. Request Name** — User-provided name for the resource request. (required)
- **b. Origination Date** — System-generated date of request entry.
- **c. Originator** — System-generated User ID of the authorized user entering the resource request.
- **d. Sponsor** — User-provided name of the individual who is to review and validate the Resource Request; the subject-matter-expert. (optional)

4 Enter Resource Definition Information into the displayed fields. Press [TAB] to move from field to field.

- **a. Activity Type** — User-provided description of the type of activity; selected by the user from a selection list of valid options. (required)
- **b. Priority** — User-provided priority for the activity. **NOTE:** Priority will be used in Release B; it is not used at this time for resource planning.
- **c. Description** — User-provided description of the activity for which the resource is required. (optional)
- **d. Resources...** — See Section 12.2.1.1, below.
- **e. Intervals...** — See Section 12.2.1.3, below. (Section 12.2.1.2 related.)

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.

- **a. Start Date** — User-provided start date of the resource request period. (required)

b. Start Time User-provided start time of the resource request. (required)

c. Stop Time User-provided stop time of the resource request. (required)

d. Stop Date User-provided stop date of the resource request. 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. (optional)

e. Frequency See Section 12.2.1.2, below. (Section 12.2.1.3 related.)

6 Enter comments concerning the resource reservation request in the **Comment** field.

7 The selections **Validated** and **Rejected** are discussed in Section 12.2.2.

8 After data is entered, click on the appropriate button(s):

Save to save data.

Clear to clear entries. Once cleared, the entries are deleted from the system.

9 **Cancel** to exit screen.

Table 12.2-1. Create a Resource Reservation - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	resource icon	double-click
2	New… button	single-click
3	resource identification information	press tab
4	resource definition information	press tab
5	duration information	press tab
6	comments	press tab
7		none
8	save data	click on Save
9	exit	click on Cancel

12.2.1.1 Selecting Resources...

Clicking on the **Resources…** button accesses a Resources screen. The **Request Name** is displayed. This screen provides a pair of lists: **Resource** 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 user selects the desired resource and, using the arrow buttons, moves the resources from one list to the other list.

1 Click on your selections and click on desired arrow.

2 Click **OK** to save your selections and exit the screen.

3 Click **Cancel** to exit the screen **without** saving changes.

12.2.1.2 Selecting Frequency

The **Intervals…** button provides the mechanism to tailor a Frequency-based request by overriding selected intervals. Clicking on **Frequency** allows the user 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. The frequency specified defaults to **Once** to indicate that the resource need covers the entire time period covered by 'Start Time' and 'Stop Time' (a single Start Date is required for this option only). Other options are identified in Table 12.2-2. The dates generated are inserted in the Selected Resources list box, described in Section 12.2.1.3, below.

12.2.1.3 Selecting Intervals...

The **Intervals…** button provides the mechanism to tailor a **Frequency-based** request by overriding selected intervals. Selecting the **Intervals…** 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 12.2.1.2, 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 Click on your selections and click on desired arrow.

2 Click **OK** to save your selections and exit the screen.

3 Click **Cancel** to exit the screen **without** saving changes.

Table 12.2-2. Frequency Qualifiers

Frequency	Text Qualifier:	Result:
Once	--	The default. Resource reservation covering the period from the start time and stop time for the start date specified.
Monthly	--	Resource reservation for every month on the start day of the month, repeated until the end date as specified.
Daily	--	Resource reservation for every day, between the start date and end date, for the start time and end time specified.
Every M-W-F	--	Resource reservation for every Monday, Wednesday, and Friday, between the start date and end date, for the start time and end time specified.
Every M-through-F	--	Resource reservation for every Monday through Friday, between the start date and end date, for the start time and end time specified.
Every T, Th	--	Resource reservation for every Tuesday and Thursday, between the start date and end date, for the start time and end time specified.
Weekends	--	Resource reservation for every Saturday and Sunday, between the start date and end date, for the start time and end time specified.
'Every'	n-days	Resource reservation for every n-days, between the start date and end date, for the start time and end time specified.

12.2.2 Edit a Resource Reservation Request

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, highlight (click on) the resource reservation request you want to modify and click on the **Modify…** button to access the Resource Reservation Request Edit screen.

3 Make the modifications. (See Section 12.2.1, above.)

4 If appropriate at this time, 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.

5 The **Status** field contains the status of the reservation request. This is system-generated based on user-input in other fields.

6 After data is entered, 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.

12.2.3 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.

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, highlight (click on) the resource reservation request you want to modify and click on the **Modify…** button to access the Resource Reservation Request Edit screen.

3 Select 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.

4 The **Status** field contains the status of the reservation request. This is system-generated based on user-input in other fields.

5 After data is entered, 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.

12.2.4 Approve a Resource Reservation Request

The **Approve** button is used when all reviews that are a part of the resource planner 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** will generate the pop-up screen again (if conflicts exist). Approval occurs after a request has been validated and the event time is acceptable.

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, highlight (click on) the resource reservation request you want to approve and click on the **Approve** button.

3 Status will show "Approved" for the selected request.

4 From the File menu, select **Exit**.

12.2.5 Commit a Resource Reservation Request

Clicking on the **Commit** button will save (Commit) all approved reservation requests so that they are accessible to Production Planning.

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, click on the **Commit** button.

3 Status will show "Commit" for the previously **Approved** requests.

4 From the File menu, select **Exit**.

12.2.6 Resource Planning Reports

Resource plans are created using approved resource requests. The resource plan is published to the Data Server for review by interested parties. This plan is passed to the Production Planning subsystem and is used for scheduling production.

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen click on the **Report** button to access the screen from which reports can be generated.

3 At the top of the window, the Resource Planner can select the report type at the ‘Report:’ selection list. There are 3 options available:

- **a.** **Ground Event Resource Usage Report.** Summarizes the percentage of resources used for different activity types.
- **b.** **Ground Event Resource Schedule.** Provides a report of the scheduled allocations of resources — ordered by resource.
- **c.** **Ground Event Resource Schedule.** Provides a report of the scheduled allocations of resources — ordered chronologically

4 In the region immediately below this, parameters needed to generate the report are input. This region is variable for each report type. The input is provided by the user through selection lists of valid options.

5 Next, there is a Destinations option in which the user selects from among the valid destination options: the Display, the printer, or a local file. If the local file destination option is selected, the 'Report File:' input region and Browse tool are used to specify where the file should be placed.
If the 'Display' option is selected as the destination, the next window region will contain the requested report.

6 Selecting **Apply** at the bottom of the screen will generate the report and send it to its destination.

7 To exit, press **Cancel**.

12.2.7 Review Resource Timeline

The Resource Planning Workbench utility allows the user to view the Resource Plan as a timeline.

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen click on the **Timeline** button.

3 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. Scroll bars allowed scrolling up and down through the full list of resources and left and right in time. A full 24-hour display is presented by clicking on the '24 hr' button at the lower left corner of the screen.

4 To exit, select **Exit** from the **File** menu.

12.2.8 Delete a Resource Reservation Request

1 Launch the Resource Reservation screen by **double-clicking** on the Resource icon located on the desktop.

2 From the Resource Planning screen, highlight (click on) the resource reservation request you want to delete and click on the **Delete** button.

3 Status will show "Deleted" for the selected request. The resource reservation request is not removed from the list 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, select **Exit** from the **File** menu.

12.3 Resource Definition Tool

The Resource Definition tool allows the authorized user to add resources or to modify the characteristics of the resources. The user can synchronize the resource planning list with the baseline set of system resources and can add or delete future resources not contained in the baseline resource list. Modifications to the resource planning list are recorded in the PDPS database. These modifications are **not** recorded in the Baseline Manager database.

The hardware resources for which resource planning can be supported include host computers, storage devices, as well as 'strings' that are made up of computers and storage devices.

12.3.1 Add a Resource

These procedures address adding a computer resource, a disk partition resource, and a generic hardware resource to the resource planning list.

1 Launch the Resource Identification screen by **double-clicking** on the Resource Definition icon located on the desktop.

2 From the Resource Identification screen, select a **Resource Type** from the list and click on the **New…** button. For Computer type, see Section 12.3.1.1. For Disk Partition type, see Section 12.3.1.2. For Hardware type, see Section 12.3.1.3. Section 12.3.1.4 defines the characteristics of a computer string, or a collection of computer type resources.

3 To exit, select **Exit** from the **File** menu..

12.3.1.1 Computer Resource Definition Screen

1 Fill in the following fields on the Computer Details screen:

Resource Name	User-defined name for the computer. (required)
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 computer. (required)
Total RAM	The total memory for the computer. (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 is used to select from when a disk is associated (or disassociated)

with the computer. 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 User comments on the resource.

2 After data is entered, click on one of these buttons:

Save to save **and** exit.

Cancel to exit **without** saving.

12.3.1.2 Disk Partition Resource Definition Screen

1 Fill in the following fields on the Disk Partition Details screen:

Resource Name User-defined name for the resource. (required)

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 kilobytes. (required)

Block Size Block size used for the disk in bytes. (required)

Comments User comments on the resource.

2 After data is entered, click on one of these buttons:

Save to save **and** exit.

Cancel to exit **without** saving.

12.3.1.3 Hardware Resource Definition Screen

1 Fill in the following fields on the Hardware Details screen:

Resource Name User-defined name for the resource. (required)

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 User comments on the resource.

2 After data is entered, click on one of these buttons:

Save to save **and** exit.

Cancel to exit **without** saving.

12.3.1.4 String Resource Definition Screen

1 Fill in the following fields on the String Details screen:

Resource Name	User-defined name for the resource. (required)
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 computers previously defined for that site. This list of computers is used to select from when a computer is associated (or disassociated) with the string. Arrow buttons will move items from this list to **Associated Computers** or from the list of **Associated Computers** to the **Computer** list.
Associated Computers	Computers in this list are associated with the string.
Comments	User comments on the resource.

2 After data is entered, click on one of these buttons:

Save to save **and** exit.

Cancel to exit **without** saving.

12.3.2 Modify a Resource

1 Launch the Resource Identification screen by **double-clicking** on the Resource Definition icon located on the desktop.

2 From the Resource Identification screen, highlight (click on) the resource you want to modify and click on the **Modify…** button to access the appropriate Details screen.

3 Make the modifications. (For field descriptions, see Sections 12.3.1.1 through 12.3.1.4, above.)

4 After data is entered, click on one of these buttons:

Save to save **and** exit.

Cancel to exit **without** saving.

12.3.3 Delete a Resource

1 Launch the Resource Identification screen by **double-clicking** on the Resource Definition icon located on the desktop.

2 From the Resource Identification screen, highlight (click on) the resource you want to delete and click on the **Delete** button. A dialogue box will pop up, asking you if you really want to delete the selections.

Click **OK**, and the resources are removed from the list and from the PDPS database and you are exited from the task. If you click on **Cancel**, you will exit without deleting.

12.3.4 Synchronize Resource Listings

This function synchronizes the Resource tables with the Baseline Manager.

1 Launch the Resource Identification screen by **double-clicking** on the Resource Definition icon located on the desktop.

2 From the Resource Identification screen, click on the **Fetch Baseline** button. This action will generate a report from the Baseline Manager to the Resource Planning workstation. The report is an ASCII listing of the production baseline.

After the report is transferred, a pop-up window appears, containing a message that the report has successfully transferred.

3 From the Resource Identification screen, click on the **Load Baseline** button. This action will synchronize the PDPS database with the baselined resource information.

4 To exit, select **Exit** from the **File** menu.

This page intentionally left blank.

13. Production Planning

The Planning Subsystem is responsible for supporting 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 these Production Requests using the **Production Workbench**. The Production Planner will issue commands to initiate plan creation and plan activation.

The Planning Subsystem submits Data Processing Requests (DPRs). A DPR will have the information needed by Processing, including Product Generation Executive (PGE)-related information; each DPR can be defined as one processing job.

13.1 Production Request Editor

From this application, the Production Planner can create new production requests, modify production requests, review data production requests, and generate reports.

The following Activity Checklist, Table 13.1-1, provides an overview of the Production Planning process. 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 13.1-1. Production Request - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launching the Production Request Editor	(P) 13.1.1	
2	Production Planner	Create New Production Request	(P) 13.1.2	
3	Production Planner	Edit/Update New Production Request	(P) 13.1.3	
4	Production Planner	Review Data Production Request	(P) 13.1.4	
5	Production Planner	Generate Reports	(P) 13.1.5	

13.1.1 Launching the Production Request Editor

In the Pre-Release B testbed, the ECS Desktop is not available. Launching the Production Request Editor is a manual operation.

Table 13.1-2 presents the steps required to launch the Production Request Editor. 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 ECS System. Enter username and password.

_ Production Planner is able to log in to the ECS System.

2 In the Terminal window, at the command line, enter **cd /usr/ecs/Rel_A/CUSTOM/bin/PLS**.

_ This is the directory containing the Planning Request Editor commands.

3 Set the display environment by entering **setenv DISPLAY ###.###.###.##:0.0** where ###.###.###.## is the IP address of the workstation being used.

_ Display environment is entered

_ See **NOTE 1**.

4 Enter **PlPREditor**.

_ The Production Request Editor is launched.

13.1.2 Create New Production Request

The new Production Request process begins when the Production Planner starts the Production Request Editor GUI using the previous procedure. The Production Planner will enter the Request Definition, PGE parameters, duration, and comments for the new Production Request.

Table 13.1-2 presents the steps required to create a new production request in a condensed format. 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 by performing procedure 13.1.1.

_ The Production Request Editor GUI is displayed.

2 Select the Production Request Editor by **single-clicking** on the **PR Edit** tab.

_ The **PR Edit GUI** page is presented.

_ See **Note 1**.

3 Enter ***username*** for the Originator name.

_ See **Note 2**.

4 Single-click the Priority field and enter the ***Priority***, then press **Tab**.

_ See **Note 3**.

5 **Single-click** the **PGE...** button.

_ The **PGE** GUI is displayed.

6 Select the desired PGE by **single-clicking** on the PGE.

_ The desired **PGE** is highlighted.

7 **Single-click** the **Ok** button.

_ The **PR Edit** GUI is displayed.

_ The following fields are automatically filled:
Satellite Name
Instrument Name
PGE Name
PGE Version

8 Select **PGE Parameters...** button.

_ The **PGE Parameters** GUI displays.

9 Select parameter to change by **single-clicking** on the **Parameter Name** row.

_ The **parameter** row highlights.

10 Enter the desired value in the **Parameter Mapping** field and press **Return**.

_ The value in the **Override Value** column is updated.

_ Repeat step for multiple fields.

11 **Single-click** on **Ok** to approve the changes.

_ The **PR Edit** GUI is displayed.

12 Enter ***MM/DD/YY*** for **Data Collection Time Range for Processing (UTC): Start: Date**, then press **Tab**.

_ See **NOTE 4**.

13 Enter ***HH/MM*** for **Start: Time:**, then press **Tab**.

14 Enter ***MM/DD/YY*** for **End: Date:**, then press **Tab**.

15 Enter ***HH/MM*** for **End: Time:**, then press **Tab**.

16 Enter **Comments**.

17 Select **File** → **Save As**

_ File selection window appears.

_ See **NOTE 5**.

18 Enter PR name in Selection field.

_ Production Request is named.

19 Select **Ok**

_ Production Request is saved.

_ Messages appear stating that the PR has been exploded into DPRs.

Note 1: 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 and then continue on the QA Monitor GUI.

Note 2: The **Origination Date** field will be automatically filled when the Production Request is saved. The PR Name field will be entered when Saving the Production Request.

Note 3: Job Priority specifies the priority of the job, with 1 (one) being the highest priority and 99 being the lowest priority. Enter priority according to DAAC policy.

Note 4: The **Duration** refers to the time range of instrument data over which the Production Request is applicable.

Note 5: Once the PR is saved, and DPRs are defined, it can not be modified. In order to correct a PR, the operator must delete the PR and re-enter the parameters.

Table 13.1-2. Create New Production Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	**double-click**
2	Select **PR Edit** tab	**single-click**
3	Enter ***originator***	press **Tab**
4	Enter **Priority**	press **Tab**
5	Select **PGE...** button	**double-click**
6	Select **PGE**	**single-click**
7	Select **Ok**	**single-click**
8	Select **PGE Parameters…** button	**single-click**
9	Select **parameter name row**	**single-click**
10	Enter **value** and press **Return**	none
11	Select **Ok**	**single-click**
12	Enter **Duration Start Date**	**MM/DD/YY,** press **Tab**
13	Enter **Duration Start Time**	**HH/MM,** press **Tab**
14	Enter **Duration End Date**	**MM/DD/YY,** press **Tab**
15	Enter **Duration End Time**	**HH/MM,** press **Tab**
16	Enter **Comments**	none
17	Exit	Click **File** → select **Save**

13.1.3 Edit/Modify Production Request

The Production planner can Edit/Modify a previously generated production request. Edit/Modification would be performed on the Priority and Duration fields. The priorities can be adjusted in response to exceptions, again based on DAAC policy. The priority updates can also occur in the Planning Workbench GUI to tasks still waiting to be submitted for processing and also in Production Monitoring using AutoSys for tasks already submitted, but not yet processed .

Table 13.1-3 presents the steps required to edit/modify a production request in a condensed format. 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 by performing procedure 13.1.1.
_ The Production Request Editor GUI is displayed.

2 Select the Production Request Editor by **single-clicking** on the **PR List** tab.
_ The **PR List GUI** page is presented.

3 **Single-click** on the PR to be Edited/Modified. Then select **File - Open**.
_ The **PR Name** row is highlighted.

4 **Single-click** on the **PR Edit** tab.
_ The **PR Edit GUI** page is presented with the selected PR.

5 Modify the ***Priority***, then press **Tab**.
_ See **Note 1**.

6 Select **PGE Parameters...** button.
_ The **PGE Parameters** GUI displays.

7 Select parameter to modify by **single-clicking** on the **Parameter Name** row.
_ The **parameter** row highlights.

8 Enter the desired value in the **Parameter Mapping** field.
_ The value in the **Override Value** column is updated.

9 **Single-click** on **Ok** to approve the changes.
_ The **Production Request** GUI is displayed.

10 Modify ***MM/DD/YY*** for **Duration Start Date**, then press **Tab**.

11 Modify ***HH/MM*** for **Duration Start Time**, then press **Tab**.

12 Modify ***MM/DD/YY*** for **Duration End Date**, then press **Tab**.

13 Modify ***HH/MM*** for **Duration End Time**, then press **Tab**.

14 Modify **Comments**.

15 Select **File** → **Save As** for a modified **Production Request.**
_ Production Request is saved.

16 Enter new PR name in Selection field.
_ Production Request is named.

17 Select **Ok**

_ Production Request is saved.

_ Messages appear stating that the PR has been exploded into DPRs.

Note 1: Job Priority specifies the priority of the job, with 1 (one) being the highest priority and 99 being the lowest priority. This procedure assumes that the Production Request has not been planned/activated.

Table 13.1-3. Edit/Modify Production Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	**double-click**
2	Select **PR List**	**single-click**
3	Select **PR Name**	**single-click**
4	Select **PR Edit**	**single-click**
5	Modify **Priority**	**enter Value**
6	Select **PGE...** button	**single-click**
7	Select desired **PGE**	**single-click**
8	Select **OK**	**single-click**
9	Select **PGE Parameters...**button	**single-click**
10	Select **Parameter**	**single-click**
11	Enter value	**none**
12	Select **OK**	**single-click**
13	Enter **Duration Start Date**	**MM/DD/YY,** press **Tab**
14	Enter **Duration Start Time**	**HH/MM,** press **Tab**
15	Enter **Duration End Date**	**MM/DD/YY,** press **Tab**
16	Enter **Duration End Time**	**HH/MM,** press **Tab**
17	Modify **Comment**	**enter text**
18	Exit	Click **File** → select **Save**

13.1.4 Review Data Production Requests

The review Data Production Request (DPR) process begins when the Production Planner starts the Planning Request Editor. The Production Planner can review DPRs associated with a PR. The Production Planner can review DPR Predicted and Actual start times, Data Start time, Status, and Priority.

Table 13.1-4 presents the steps required to review DPRs in a condensed format. 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 by performing procedure 13.1.1.

_ The Production Request Editor GUI is displayed.

2 **Single-clicking** on the **DPR List** tab.
_ The **DPR List** GUI is presented.

3 **Single-click** the **Production Request** pull-down arrow.
_ A list of Processing Requests is displayed.

4 **Single-click** the **Production Request** for DPR listing.
_ The **Production Request** is entered into the field.

5 **Single-click Filter**.
_ The list of DPRs is displayed.

6 **Single-click** the **DPR View** tab.
_ The **DPR View** GUI is displayed.

7 **Single-click File - Open**.
_ The **PGE Parameters** GUI displays.

7 **Single-click** DPR from list.
_ DPR appears in **Selection** field.

8 **Single-click OK**.
_ The **DPR** information is displayed.

9 **Single-click** the **PGE Parameters...** button.
_ The **PGE Parameters** GUI displays.

10 **Single-click** the **Ok** button.
_ The **DPR View** GUI is displayed.

11 **Single-click** the **PGE File Mappings...** button.
_ The **PGE File Mappings** GUI displays.
_ Input Data and Output Data tables are displayed.

12 **Single-click** on **Ok**.
_ The **DPR View** GUI is displayed.

13 Review **DPR Status**.
_ **Predicted Start Date** and **Time** is displayed.
_ **Actual Start Date** and **Time** is displayed.
_ **Priority** and **Status** are displayed.
_ See **Note 1**.

14 Repeat steps 2 through 13 to review multiple **DPR**s of multiple **PR**s.

Note 1: **Predicted** and **Actual Start Date** and **Times** and **Status** will not be displayed if the Production Request has not been scheduled.

Table 13.1-4. Review Data Production Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	**double-click**
2	Select **DPR List**	**single-click tab**
3	Enter **PR *name***	**single-click Filter**
4	Select **DPR Id**	**single-click DPR View**
5	Select **PGE Parameters...**	**single-click**
6	Select **OK**	**single-click**
7	Select **PGE File Mappings...**	**single-click**
8	Select **OK**	**single-click**
9	Review **Data Start Time and Data Stop Time**	none
10	Review **DPR status**	none
11	Repeat 2 through 9 to review DPRs	

13.1.5 Reports

The generate Reports process begins when the Report Generator GUI is launched by a member of the Production Team. A selection of eleven standard reports is available.

Processing Status Report - Provides status of production requests (PRs) for a specified instrument and a specified period. PRs are included in the report if they had Data Processing Requests (DPRs) run during the reporting time period, or if PRs are currently active within the DAAC plan. The reporting periods for which status is determined are 1 day (current) and 3, 7, 30, and 90 days in the past. The 1, 3, 7 day reports are updated daily, while the 30 and 90 day reports are updated weekly. One set of reports is generated per instrument per DAAC. This report is generated automatically once per day and accessible via the Document Data Server.

Processing Errors Report - Provides error status from all failed DPRs during one day grouped by PR, and summaries for the preceding 3, 7, 30 and 90 days. This report is produced daily, one report per instrument per DAAC. This report is generated automatically once per day and accessible via the Document Data Server.

Resource Usage Report - The report summarizes the resource usage for the day. In addition, comparisons of the average resource usage from the successful DPRs for the preceding 7 and 30 days are also produced daily. This report is generated automatically once per day and accessible via the Document Data Server.

Actual vs. Plan Report - The report compares the active production plan with the current processing status at the product group, product, and PGE level, and gives account of production backlogs. This report is only generated on demand by operations.

Ground Event Resource Utilization - This report summarizes the resource usage by DPR (PGE run) for a given day. In addition, it provides for comparison the average of resource usage from the successful PGE runs for the preceding 7 and 30 days. This report is run daily, stored on the ECS Document Data Server. One report is generated per instrument per DAAC.

Ground Event Resource Schedule Report (by resource) - The report summarizes the scheduling of resources. This report is only generated on demand by operations.

Ground Event Resource Schedule Report (Chronological) - The report summarizes the scheduling of resources. This report is only generated on demand by operations.

Job Report - This report supplies local operations staff with a detailed view of all data production jobs currently tracked in the data production environment. By default, the report displays all jobs, but can be restricted to show only jobs in a particular state. This report is only generated on demand by operations.

Dependency Jobs Report - This report provides local operations with information on the jobs that depend upon a particular DPR which the operator is prompted to enter. This report is only generated on demand by operations.

Disk Availability Report - The report provides the operations with an overview of the disk space available on the processing platforms. This report is only generated on demand by operations.

Production Plan Report - This report provides a text version of the timeline view produced by the planning subsystem. The report lists planning information for each DPR organized by PR. This report is stored on the Document Data Server.

13.1.5.1 Generate Standard Production Reports

Table 13.1-5 presents the steps required to generate a Production Reports in a condensed format. Not all reports are intended to be generated at one time. Some reports, as mentioned in the above text, are generated automatically and others are generated only on demand by an operator. 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** GUI by double-clicking on the Icon located on the desktop.

_ The **Production Request Editor** GUI is displayed.

2 Select **File** → **Generate Report** GUI by double-clicking on the Icon located on the desktop.

_ The **Report Generator** GUI is displayed.

3 **Single-click** and hold on the **Report** field.

_ A list of reports appears:

_ **See Note 1.**

4 Highlight the desired report by moving the cursor to the desired report and release the mouse button.

- Selected report appears in the **Report** field.
- Report fields appear in the Report selection area.

5 This step varies depending on type of report selected. Select/enter the **Instrument, Days, Start Date, End Date, DPR I.D., Type of Plan, and/or Plan Name** as required by the specific report.

- **Instrument**, **Days** and **Type of Plan** are pull down selections.
- **Start Date**, **End Date**, **DPR I.D.**, and **Plan Name** require operator input.
- Some reports require no input.

6 **Single-click** CheckButtons that are desired in the **Destination** area.

- **Display** is the default and always selected.
- **Printer** will send the file to the selected printer.
- **File** will save the file to your selected directory.

7 **Enter Report File destination**.

- A list of destinations appears: **Display**, **Printer**, **Data Server**, and **File**.

8 **Single-click** the **Apply** button.

- The report is generated.
- Report appears in the Report area of the GUI.
- File is printed, if selected in step 5.
- File is saved to Document Data Server, if selected in step 5.
- File is saved to local directory, if selected in step 5.

9 Repeat steps for desired reports.

- Other reports are generated.

10 **Single-click** the **Cancel** button.

- **Generate Report** GUI is closed.

NOTE 1: Processing Status, Processing Errors, and Ground Event Resource Usage Reports are generated automatically.

Table 13.1-5. Generate Reports - Quick-Step Procedures

Report	Data Entry Fields
Processing Status Report	Instrument, Days
Processing Errors Report	Instrument, Days
Resource Usage Report	Instrument
Actual vs. Plan Report	Instrument, Days
Ground Event Resource Utilization	None
Ground Event Resource Schedule Report (by resource)	Start/End Dates
Ground Event Resource Schedule Report (Chronological)	Start/End Dates
Job Report	None
Dependency Jobs Report	DPR ID
Disk Availability Report	None
Production Plan Report	Type of Plan, Plan Name

Table 13.1-6. Generate Reports - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	**double-click**
2	Select **File** → **Generate Report**	**single-click pull-down list**
3	Select **Report**	**single-click pull-down list**
4	Select **Report type**	**highlight and release mouse button**
5	Select/Enter required report parameters.	**Enter parameters or select on pull-down list as required**
6	Select **Destination**	**single-click pull-down list**
7	Enter **File destination**	**single-click**
8	Generate **Report**	**single-click Apply**
9	Exit	**single-click Cancel**

13.2 Production Planning Workbench

From the Production Planning Workbench application, the Production Planner can create Baseline, Candidate and Activate production plans and display a planning timeline. This GUI is used to schedule the Production Requests generated by the Production Request Generator.

The following Activity Checklist, Table 13.2-1, provides an overview of the Production Planning process. 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. Production Planning - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launching the Planning Workbench	(P) 13.2.1	
2	Production Planner	Generate/Baseline/Activate a Production Plan	(P) 13.2.2	
3	Production Planner	Review Plan Timeline	(P) 13.2.3	

13.1.1 Launching the Planning Workbench and Planning Timeline

In the Pre-Release B testbed, the ECS Desktop is not available. Launching the Planning Workbench is a manual operation.

Table 13.1-2 presents the steps required to launch the Planning Workbench. 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 ECS System. Enter username and password.

_ Production Planner is able to log in to the ECS System.

_ See **NOTE 1**.

2 In the Terminal window, at the command line, enter **cd /usr/ecs/Rel_A/CUSTOM/bin/PLS**.

_ This is the directory containing the Planning Request Editor commands.

3 Set the display environment by entering **setenv DISPLAY ###.###.###.##:0.0** where ###.###.###.## is the IP address of the workstation being used.

_ Display environment is entered.

4 Enter **st_all.gsfc.csh***.

_ The Message Handler is launched.

4 Enter **st_plwb.autosys.gsfc.csh***.

_ The Planning Workbench is launched.

5 Enter **st_pltl.gsfc.csh***.

_ The Planning Timeline is launched.

Note 1: If doing a **telnet** into **plng1sun**, the operator must do a **xhost +** in order to gain access to the GUI.

13.2.1 Create New Production Plan

The new Production Plan process begins when the Production Planner starts the Planning Workbench. The Production Planner will perform the high level functions, such as creating a plan, deciding when to activate, cancel, and replan. Job priority may also be changed.

The Production Planner will bring the plan back on track due to late data arrivals, QA issues, equipment failures and other events can cause significant delays in the processing of the associated products.

Table 13.2-1 presents the steps required to create a new Production Plan in a condensed format. 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 by performing procedure 13.2.1.

_ The **Planning Timeline** GUI is displayed.

_ See **Note 1**.

2 Select **File** → **New**.

_ The New window appears.

3 Enter the *Plan Name* in the Plan Name field**,** then **single-click Ok**.

_ Name is displayed in Plan Name Field.

_ Status is **Candidate**.

4 Single-click inside the **Comment** field and enter ***Comments*** .

_ See **Note 2**.

5 Select the desired Production Request by **single-clickin**g on the Request and **single-clicking** on **Schedule/Unscheduled** up and down arrows.

_ Production Request will appear in appropriate panel.

_ See **Note 3**.

_ See **Note 4**.

6 Select Production Request and **single-click Prioritize.**

_ Priority window appears with PR priority shown.

7 Enter new priority, if desired, and **single-click Ok.**

_ Priority is updated in table.

_ See **Note 5**.

8 Select a PR in the **Scheduled** window and then **single-click View DPRs .**

_ PR is highlighted when selected.

_ **DPR View** window is displayed.

9 Exit **View DPRs** by selecting O**k.**

10 Select **File** → **Save**

_ Production Plan is saved.

11 If desired, select **Activate** to activate the new Plan.

_ The **End Date** window is displayed.

_ See **Note 6.**

10 Enter **End Date** and time and select **Ok**.

_ Production Plan is activated.

12 If desired, select **Baseline** to a Baseline a Plan and select **Ok**..

_ **The current plan is "*planname*". Do you wish to baseline it?** window appears.

13 If desired, select **Kill** to kill a Plan and select **Ok**.

_ **Do you want to kill all jobs in the AUTOSYS queue?** window appears.

14 Select **File - Quit** to exit the Planning Workbench.

Note 1: If the current plan is running, the following message may appear. "The Production Request "xxx" contains no DPRs which have not been activated."

Note 2: Operator may use the "sash" button (the line separating the area between the comment filed and the scheduling PR area with a square box on the right hand side) to expand or contract the text area.

Note 3: Operator can single-click on multiple Production Requests to add all at the same time.

Note 4: If the PR depends upon another PR that is not scheduled, then the message "Production Request "xxx" must be selected before DPR "yyy" of Production Request "zzz" can be scheduled. If a PR has been produced, it can not be rescheduled.

Note 5: Enter a priority between 1 and 99, with 1 being the highest priority. Enter priority according to DAAC policy.

Note 6: Rollover time. This area is automatically filled when a new plan is activated.

Table 13.2-2. Create New Production Plan - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Workbench**	**double-click**
2	**File → New**	**single-click File pull-down**
3	Enter **Plan *name***	press **Tab**
4	Enter **Rollover Time**	**MM/DD/YY**
5	Enter **Comments**	press **Tab**
6	Select/Deselect **Production Requests**	**single-click up and down arrow buttons**
7	**File → Save**	**single-click File pull-down**
8-10	Select **Baseline, Activate, or Kill**	**single-click Push Button**

13.2.2 Review Plan Timeline

The new Production Plan process begins when the Production Planner starts the Planning Workbench from the normal Production Planner's desktop. Table 13.2-2 presents the steps required to review Planning Timelines in a condensed format. 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 Timeline by performing procedure 13.2.1.

_ The **Planning Timeline** GUI is displayed.

2 Select **Plans** from the menu bar.

_ List of Plans appears.

3 Select the desired Plan to review.

_ The timeline for the specific plan is displayed.

_ Name is displayed in the Title bar.

4 Select **Time - Change Plan Window**.

_ Window appears with default times.

5 Enter **Plan Win Start** date and time.

_ mm/dd/yy and hh/mm are displayed.

6 Enter **Plan Win End** date and time and select **Ok**.

_ mm/dd/yy and hh/mm are displayed.

7 **Single-click** the **Show** pushbutton and select one of 12 time increment between **5 min** to **64 hrs** for timeline scale.

_ See **Note 1**.

8 Select Display - Change Colors.

_ Color grid appears with a list of states.

9 **Single-click** state and **single-click** desired color for that state.

_ Color is changed for that state.

_ New color appears on the horizontal bar between color and state selections.

10 Select **Display - Change Resources**.

_ **Available** and **Viewed Resources** list appear.

11 **Single-click** resource in desired list and select Add or Del.

_ **Add** will move the list from **Available** to **Viewed** list.

_ **Del** will only remove items from the **Viewed** List.

12 **Single-click** on an item in the **View Resource** List.

_ Item is highlighted.

13 **Single-click** on the **up** or **down** arrow.

_ Resource selected will move up or down in order on the list.

5 ???.

_ ???.

5 ???.

_ ???.

5 ???.

_ ???.

Note 1: The entry "**OTHER**" has no purpose at this time.

Note 2:

Table 13.2-3. Review Plan Timeline - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1		
2		
3		
4		
5		
6		

14. Production Processing

The Data Processing Subsystem provides a batch processing environment to support the generation of data products. It manages, queues, and executes Data Processing Requests (DPR) on the processing resources at a DAAC. A DPR can be defined as one processing job. Each DPR encapsulates all of the information needed to execute this processing job. DPRs are submitted from the Planning Subsystem, which in turn is triggered by the arrival of data or internally through Planning itself. DPRs use Product Generation Executives (PGEs) to perform processing. PGEs will result from the integration and test of delivered science algorithms and also user-specific methods in the subsystem. They will be encapsulated in the ECS environment through the SDP Toolkit. The Data Processing subsystem provides the Operational interfaces needed to monitor the execution of science software PGEs.

The following Activity Checklist, Table 14.1-1, provides an overview of production processing. 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 14.1-1. Production Planning - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Monitor	Configure AutoSys	(P) 14.1	
2	Production Monitor	Review Hardware Status	(P) 14.2	
3	Production Monitor	Review DPR Dependencies	(P) 14.3	
4	Production Monitor	Review DPR Production Timeline	(P) 14.4	
5	Production Monitor	Modify Job Priority	(P) 14.5	
6	Production Monitor	Review Alarms	(P) 14.6	
7	Production Monitor	Review Job Activities	(P) 14.7	
8	Production Monitor	Modify Job Status	(P) 14.8	
9	Production Monitor	Display Activity Log	(P) 14.9	
10	Production Monitor	Display and Print Job Dependency Log	(P) 14.10	
11	Production Monitor	Define Monitors/Browsers	(P) 14.11	
12	Production Monitor	Modify Database Maintenance Time Change	(P) 14.12	
13	Production Monitor	Generate Production Reports	(P) 13.7 & (P) 14.13	
14	Production Monitor	Time Synchronization	(P) TBD	

14.1 Configure AutoSys

The configure AutoSys process begins when the Production Monitor starts AutoSys and changes Runtime Options or uses the vi editor to modify configuration files.

The procedures in this section concern configuring AutoSys and hardware groups.

14.1.1 AutoSys Runtime Options

The Runtime Options that may be defined by the Production Monitor operator are:

Refresh Interval - The Refresh Interval is how often the View display will be updated.

Ping Interval - The Ping Interval is defined by how often the connectivity will be evaluated.

Hang Time - The Hang Time is the length of time jobs will continue to display within a machine after they have completed running.

Inches/Hour- The Inches/Hour is how much information is displayed on the screen. All values are initially set to default values by the AutoSys system.

Table 14.1-2 lists the Runtime options available for HostScape, TimeScape, and JobScape. Not all options are available for all graphical user interfaces (GUI).

HostScape displays jobs on a machine-by-machine basis, indicating which AutoSys server/client machines are up and active, and which jobs are running or have recently run on each machine. This interface is used to monitor hardware status in real-time.

TimeScape presents a Gantt-like view of a job processing from a temporal (or time-related) point-of-view. This interface depicts all job types: Command Jobs, Box Jobs, and File Watcher Jobs. It also depicts the nesting of jobs within boxes and the duration of time it will take for jobs to complete. This interface is used to monitor job flow in real-time.

JobScape presents a Pert-like view of job processing from a logical (or job dependency) point of view. This interface depicts all job types: Command Jobs, Box Jobs, and File Watcher jobs. It also depicts the nesting of jobs within boxes and the dependencies between jobs. This interface can be used to monitor job flow in real-time.

Table 14.1-2. Runtime Options Table

Interface	Refresh Interval	Hangtime	PING	Inches/Hour
HostScape	X	X	X	
TimeScape	X			X
JobScape	X			

T o configure AutoSys use the following procedures. Perform only those steps applicable to the interface, as defined in Table 14.1-2.

1 Launch **AutoSys** by **double-clicking** on the **AutoSys** Icon located on the desktop.

_ The **AutoSys** GUI is displayed.

2 **Single-click** on either **HostScape**, **TimeScape**, or **JobScape**.

_ The desired **GUI** dialog box is displayed.

3 Execute menu path: **Options** → **Edit Runtime Options**.

_ The **Runtime Options** dialog box is displayed.

4 **Single-click Refresh Interval (Seconds)**, and enter a value between **1** to **99999**.

_ Value is entered.

_ Default value is **30**

_ **Reloading Job Data** window reappears every ## seconds.

If Freeze Frame feature is enabled, changes will not take place until it is disabled.

5 **Single-click Ping Interval (Seconds)** and enter a value between **1** to **99999.**

_ Value is entered.

_ Default value is **300**

_ 99999 means no **ping** commands are issued.

If Freeze Frame feature is enabled, changes will not take place until it is disabled.

6 **Single-click Hang Time (Minutes)** and enter a value between **1** to **99999**.

_ Value is entered.

_ Default value is **1**

If Freeze Frame feature is enabled, changes will not take place until it is disabled.

7 **Single-click Inches/Hr (inches)** and enter a value between **1** to ###.

_ Value is entered.

_ Default value is **2**

If Freeze Frame feature is enabled, changes will not take place until is disabled.

8 **Single-click Apply**.

_ The Runtime Options are set.

9 **Single-click OK**.

_ The dialog box closes.

14.1.2 Configure Hardware Groups

This section explains how to configure AutoSys hardware groups. The default group is "All Machines." If the Production Monitor requires to monitor a specific set of machines to be monitored, groups may be defined.

To configure AutoSys hardware groups use the following procedures. (The example for these procedures is HostScape.)

1 Access the command shell.

_ The Command shell **$** is displayed.

2 Type **AUTOUSER/xpert.groups.$AUTOSERV**.
A blank page is presented.

3 Enter "**groupname**: ***groupname***" and press **Return**.
(Repeat step 3 for each item in the group.)

Repeat steps 2 and 3 for additional groups.

4 Enter "**localhost**" and press **Return**.

5 Enter "***machine name***" and press **Return.**

6 Type **:w** to save the file.

```
Groupname:  Modis
localhost
Tabby
Tomcat
Groupname: Archive
localhost
Sheltie
Doberman
```

Figure 14.1-1. AutoSys Hardware Group File Example

7 Launch the **AutoSys** and **double-click HostScape**.

_ The **HostScape** GUI page is presented.

8 Select **View** → **Select Machine Group.**

_ The **Machine Group Selection** dialog box is presented.

9 Select ***machine group*** to be presented

_ The **Machine Group** is highlighted.

10 **Single-click Apply** PushButton**.**

_ The selected **machine group** is applied.

11 **Single-click OK** PushButton**.**

_ The **Machine Group Selection** dialog box is closed.

14.2 Review Hardware Status

The review hardware status process begins when the Production Monitor starts the AutoSys from the Production Monitor's desktop. Hardware status is performed through the AutoSys HostScape GUI, which is used to monitor hardware status in real-time.

14.2.1 Review Hardware Status

To review hardware status using AutoSys HostScape use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon..

_ The **AutoSys** GUI is displayed.

2 **Single-click** on the **HostScape** PushButton.

_ The **HostScape** GUI page is presented.

_ View presented is **Normal** View.

3 Review the Control Region (left side of display) to identify color code for status of machines. This code will be displayed on the machine box border in the **View Region**.

_ **MACHINE UP** (active) is Green.

_ **MACHINE DOWN** (inactive and cannot be reached) is Red.

_ Machine Inactive is Black**.** (Not shown in Control Region)

4 Review machine type in **View Region**.

_ The ***machine name*** is displayed.

_ Server machines are in the first (top) row of the display.

_ Event Server name appears below list of jobs, if applicable.

_ Event Processor name appears below list of jobs, if applicable.

_ Client machines are in the subsequent rows of the display.

5 Review machine boxes in the View Region to ascertain status of individual machines.

_ The total number of jobs STARTING or RUNNING.

_ All jobs RUNNING are listed.

6 Review the **Alarm** indicator/PushButtons of individual machines in the View Region.

_ Red indicates an alarm has been generated.

_ Gray (default color) indicates normal operation.

If an Alarm is present, **single-clicking** alarm buttons will bring up the Alarm Manager

7 Review machine connection status in the View Region.

_ Solid black line indicates AutoSys can communicate with the client machine Internet daemon.

_ Solid red line indicates AutoSys cannot communicate with the client machine Internet daemon; however, the daemon does respond to **ping** commands.

_ Dashed red line indicates AutoSys cannot communicate with the client machine; machine is probably turned off.

8 Exit **Hostscape.**

a Select **File** → **Exit**.
HostScape will quit.

14.2.2 Hardware Status View Options

The View Options provide three methods to view the hardware status:

The Normal view (default) displays three rows of machines with job activities.

The Global view displays seven rows of machines but not job activities.

The Zoom view displays one machine with great detail: Job name, description, status, and commands.

The Performance Monitor operator may select the Global view to monitor the entire system and in the case of a malfunction, use the Zoom view to focus on the specific problem machine.

To view hardware status from HostScape use the following procedures:

1 Select a machine by **single-clicking** on its ***name*** and executing menu path: **View** → **Select View Level** → **Global View**.

_ The **Global** view is displayed.

_ Seven rows of machines are displayed.

_ No job information is displayed.

2 Select a machine by **single-clicking** on its ***name*** and then execute menu path: **View**→ **Zoom in Machine**.

_ The **Zoom** view is displayed.

_ A table of **Job Name**, **Description**, **Status**, and **Commands** is displayed.

3 Select **Dismiss**.

_ The **Global** view is displayed.

4 Execute menu path: **View** → **Select View Level** → **Normal view**.

_ The **Normal** view is displayed.
_ Three rows of machines are displayed.
_ Limited job information is displayed.

14.3 Review DPR Dependencies

The review DPR dependencies process begins when the Production Monitor starts the AutoSys Jobscape from the Production Monitor's desktop. The JobScape interface is used to monitor job flow in real-time.

To review DPR Dependencies from the JobScape mode use the following procedures:

1 Launch the Production Planning Workbench by **double-clicking** on the **AutoSys** Icon located on the desktop.

_ The **AutoSys** GUI Control Panel is displayed.

2 **Single-click** on the **JobScape** PushButton.

_ The **JobScape** GUI page is presented.

3 Review the Control Region (left side of display) to identify **True** or **False** Dependency Legend.

_ **True** (default **solid** arrow) indicates job dependencies have been met.
_ **False** (default **dashed** arrow) indicates job dependencies have **not** been met.

Dependency arrows only indicate that a job dependency exits for a job. They do not define time-related starting conditions, nor do they describe the type of job dependency, such as success, failure, or running.

4 Review the Job Display for status. Default colors are:

_ White indicates job status of **ACTIVATED**.
_ Black indicates job status of **INACTIVE** or **ON-HOLD** or **ON-ICE**.
_ Yellow indicates job status of **QUE WAIT**.
_ Orange indicates job status of **RESTART**.
_ Forest Green indicates job status of **STARTING** or **RUNNING**.
_ Red indicates job status of **FAILURE** or **TERMINATED**.
_ Light Blue indicates job status of **SUCCESS**.

5 Review the Job Display for job types:

_ **Rectangle** depicts **Box Jobs**.
_ **Ellipses** depicts **Command Jobs**.
_ **Hexagons** depicts **File Watcher Jobs**.

6 Select a job by placing the **cursor** on a job and pressing the **left** mouse button.

_ Border around selected job changes to **yellow**.
_ Job name appears in Current Job Name area of the Control Region.

7 Review job descendants by placing the **cursor** on a job and pressing the **right** mouse button.

- **Descendants** popup menu appears.
- Border around selected job changes to **yellow**.
- Job name appears in Current Job Name area of the Control Region.

8 Select **Show Children** on the **Descendants** popup menu.

- Job's first level **Command, File Watcher,** and **Box Jobs** appear.
- Repeat step 6 to change job selection.

9 Select **Show All Descendants** on the **Descendants** popup menu.

- Job's Command, File Watcher, and Box Jobs appear for all levels.

10 Select **Hide All Descendants** on the **Descendants** popup menu.
Default view is displayed.
All dependents are hidden.

11 Exit **JobScape.**

a Select **File** → **Exit**.
JobScape will quit.

14.4 Review DPR Production Timeline

The review DPR Production Timeline process begins when the Production Monitor starts AutoSys TimeScape from the Production Monitor's desktop. The TimeScape interface is used to monitor job flow in real-time.

To review the DPR Production Timeline from TimeScape use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon.

- The **Production Planning** GUI is displayed.

2 **Single-clicking** on the **TimeScape** button.

- The **TimeScape** GUI page is presented.
- Current time is displayed in red.

3 Review **Actual/Projected** Legend in lower left of the **Control Region** and compare to **View Region**.

- **Projected** is a rectangular (blue filled) graphic, to show average job completion time.
- **Actual** is a striped (white and blue) ribbon, to show how much of the job has completed.
- If stripe is green, job is running.
- If stripe is black, job has completed.

4 Review job descendants by placing the **cursor** on a job and pressing the **right** mouse button.

_ **Descendants** popup menu appears.

An Asterisk (*****) indicates that a Box Job's descendants have been hidden.

5 Select **Show Children** on the **Descendants** popup menu.

_ Job's first level **Command, File Watcher,** and **Box Jobs** appear.

6 Select **Show All Descendants** on the **Descendants** popup menu.

_ Job's **Command**, **File Watcher**, and **Box Jobs** appear with all levels.

7 Select **Hide All Descendants** on the **Descendants** popup menu.

_ Default view is displayed.

8 Exit **TimeScape**

a Select **File** → **Exit**.
TimeScape will quit.

14.5 Modify Job Priority

The modify job priority begins when the Production Monitor starts AutoSys from the Production Monitor's desktop.

Job Priority specifies the priority of the job, with 1 (one) being the highest priority. Priorities have a "Band" associated with them; i.e., 1-99 is band "0", 100 - 199 is band "1", etc. Higher band jobs (lower numerically) completely block lower band jobs, preventing a situation where a high-priority, resource intensive job cannot obtain enough resources.

To configure AutoSys, use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon..

_ The **AutoSys** GUI is displayed.

2 **Single-click** on the Job Definition button.

_ The **Job Definition** GUI page is presented.

3 **Single-click** on the **Adv Features** button.

_ The Job Definition Advanced Features GUI page is presented.

4 In the UNIX Command Information area, **single-click** and modify the **Que Priority** to the desired value.

_ **Que Priority** value is updated.

CAUTION

Do not modify other fields on the **Job Definition Advanced Features** GUI. ECS cannot disable these GUI features, because AutoSys/AutoXpert is a COTS product. Training and documentation will be provided to operations personnel, which will provide clear instructions concerning the intended use and which features to avoid.

5 **Single-click** on the **Save&Dismiss** PushButton.

_ The **Job Definition** GUI page is presented.

6 **Single-click** on the **Exit** PushButton.

_ The **Job Definition** GUI page is presented.

7 **Single-click** on the **Exit** PushButton.

_ The **Auto JAC** GUI page is presented.

8 **Single-click** on the **Ok** PushButton.

_ AutoSys is exited.

14.6 Review Alarms

The review alarms process begins when the Production Monitor starts AutoSys Alarm Manager from the Production Monitor's desktop. The Alarm Manager allows the Production Monitor to view alarms as they arrive, provide a response, and change the alarm status. The Alarm Manager is also configurable for the types of alarms that get displayed.

14.6.1 Review Alarms

To use AutoSys Alarm Manager to review alarms, use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon located on the desktop.

_ The **AutoSys** GUI is displayed.

2 **Single-click** on the **Ops Console** Icon located on the desktop.

_ The **Ops Console** GUI is displayed.

3 **Single-click** on the **Alarm** PushButton.

_ The **Alarm Manager** GUI page is presented.

Alarms are displayed in reverse order of occurrence; the newest alarm appears at the top of the list.

4 **Single-click** on an alarm in the **Alarm List**.

_ Information for **Alarm Type**, **Job Name**, **Time**, **State**, **Comment** is displayed.

_ Alarm is displayed in detail in the **Currently Selected Alarm** region of the display.

5 **Single-click** the **Response** edit box and enter ***response***, if desired.

_ Response is entered.

6 Update **Alarm State** by **single-clicking** proper radio button of **Open**, **Acknowledged**, or **Closed**.

_ Alarm State is updated.

7 **Single-click Apply**.

_ Response is entered.

8 **Repeat steps 4 - 7 to update/review multiple alarms**.

_ Alarms are updated/reviewed.

9 **Single-click Ok**.

_ ???.

10 Exit **Alarm Manager.**

a Select **File** → **Exit**.
Alarm Manager will quit.

14.6.2 Alarm Selection Configuration

To configure the AutoSys Alarm Manager to control which alarms are displayed, use the following procedures:

1 Execute menu path: **View** → **Select Alarms**.

_ **Alarm Selection** GUI is displayed.

Alarm Selection defaults are

All Types for **Select by Type**,

Open and **Acknowledge** for **Select by Stated**, and

All Times for **Select by Time**.

2 To select a single alarm, **single-click** on an alarm in the **Select by Type** list or select **All Types**.

_ Alarm types are selected.

_ If **All Types**, button will turn yellow.

3 To select multiple alarms: **press and hold** the **Control** key while **single-clicking** alarms in the **Alarm List**.

_ Multiple alarms are selected.

4 To **Select by State**, **single-click** on appropriate **Open**, **Acknowledge**, **Closed**, or **All States** ToggleButtons.

_ Any or all buttons can be selected.

_ Button will turn yellow when selected.

5 To **Select by Time**, enter **From Date (MM/DD/YY)** and press **Tab,** or select **All Times**.

_ **MM/DD/YY** is entered.

_ If **All Times**, proceed to step 9.

6 Enter **From Time (hh:mm),** and press **Tab**.

_ **hh:mm** is entered.

7 Enter **To Date (MM/DD/YY),** and press **Tab**.

_ **MM/DD/YY** is entered.

8 Enter **To Time (hh:mm),** and press **Tab**.

_ **hh:mm** is entered.

9 Select **Apply**.

_ Selections are applied.

10 Select **OK**.

_ **Alarm Selection** GUI is closed.

_ **Alarm Manager** GUI is displayed.

11 Select **Options** → **Sound On**.

_ **Sound On** Toggle button appears yellow when sound is on.

12 Exit **Alarm Manager.**

a Select **File** → **Exit**.
Alarm Manager will quit.

14.7 Review Job Activities

The review Job Activities process begins when the Production Monitor starts AutoSys from the Production Monitor's desktop. The Job Activity Console is the primary interface that allows the operator to monitor all jobs that are defined to AutoSys. The Job Selection GUI sets the criteria for Job Status and Machine to display jobs.

14.7.1 Review Job Activities

To review Job Activities using AutoSys, use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon..

_ The **AutoSys** GUI is displayed.

2 **Double-click** on the **Job Console** PushButton.

_ The AutoSys Job Activity Console is presented.

3 Review the **Job List** region.

_ **Job Name, Description, Status, Commands,** and **Machine** are displayed in a table.

4 **Single-click** anywhere on a job row to display detailed information.

_ Job details displayed in **Currently Selected Job** region.

5 Review **Starting Conditions**.

_ Job **Starting Conditions** display.

_ **Atomic Condition, Current State,** and **T/F** are displayed.

Single-clicking on a Starting Condition will update the Currently Selected Job to those "upstream" dependencies.

6 Review **Report Area**. **Single-click** on **Summary**, **Event**, or **None** in the **Reports** area to view different reports.

_ **Summary**, **Event**, and **Job Reports** display as selected.

_ Select report button turns yellow.

7 **Single-click Exit**.

_ **AutoSys JAC Exit** GUI appears.

8 **Single-click OK**.

_ **AutoSys Job Activity Console** GUI is quit.

14.7.2 Review Job Selection Criteria

To review Job Selection Criteria use the following procedures.

1 Execute menu path: **View** → **Select Jobs**.

_ The **Job Selection** view is displayed.

Job Selection defaults are **All Jobs** (Job Name) for **Select by Name**

All Statuses for **Select by Status**

All Machines for **Select by Machine**

Unsorted for **Sort Order**.

2 **Single-click** anywhere on a job row to display detailed information.

_ Job details display in **Currently Selected Job** region.

3 **Single-click** desired option in the **Select by Name** area, and enter required name or select **All Jobs**.

_ Options are **Job Name, Box Name, or Box Levels or All Jobs**.

_ Selection button turns yellow.

4 **Single-click** desired status in the **Select by Status** area

_ Options are **Starting, Running, Success, Failure, Terminated, Restart, Que Wait, Activated, Inactive, On Hold,** and **On Ice**.

5 **Single-click** desired machine in **Select by Machine** area or select **All Machines**.

_ Machine is highlighted.

_ **All Machines** button turns yellow.

6 **Single-click** desired Sort Order.

_ Options are **Start Time, End Time, Job Name, Job Status, Machine Name,** and **Unsorted**.

7 **Single-click Apply** .

_ Selections are applied.

8 **Single-click OK** .

_ **Job Activity Console** is displayed.

_ **Job List** displays in accordance with new selection criteria.

9 **Single-click Exit**.

_ **AutoSys JAC Exit** GUI is displayed.

10 **Single-click Ok**.

_ **Job Activity Console** GUI is closed.

14.8 Modify Job Status

The modify Job Status process begins when the Production Monitor starts AutoSys from the Production Monitor's desktop.

To modify Job Status using **AutoSys Send Event** GUI, use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon..

_ The **AutoSys** GUI is displayed.

2 **Double-click** on the **Ops Console** Icon.

_ The **AutoSys Job Activity Console** is presented.

3 **Single-click Send Event** button in the **Actions** Region.

_ **Send Event** GUI displays.

Send Event defaults are:

Start Job for **Event Type**

Now for **Time**

Normal for **Send Priority**.

4 **Single-click Event Type** to select desired status.

_ Options are **Start Job, Job On Hold, Job Off Hold, Comment, Force Start Job, Job On Ice, Job Off Ice, Stop Demon, Kill Job, Change Status,** and **Change Priority**.

5 **Verify Job Name**.

_ ***Job Name*** appears in field.

6 **Enter** desired Date and Time, either **Now** or **Future**. **Single-click Now** or **single-click Future** and enter **Date** (mm/dd/yy), **Time** (hh:mm), and **single-click A.M** or **P.M**.

_ **Now** for immediate execution.

_ **Future** for future time and date. (Current date and time are default values.)

7 **Enter *comment.***

_ Free form field for text entry to associate with event.

8 Review **AUTOSERV Instance** field**.**

_ Machine on which job will run. The Production Monitor may specify another machine by entering that machine name.

9 Review **Change Status** option menu**. Single-click** and select from pop-up list desired status.

_ Options are: **Running**, **Success**, **Failure**, **Terminated**, **Starting**, and **Inactive**.

_ Can be changed only if **Change Status** was selected in the Event Type region.

10 Review **Queue Priority** entry**.**

_ Can be changed only if **Change Priority** was selected in the **Event Type** region.

11 Review **Send Priority** RadioBox**.**

_ Can be changed only if **Change Status** was selected in the **Event Type** region.

_ High priority is reserved for emergencies.

12 **Single-click Execute**.

_ **Send Event** setting is set.

_ **Job Activity Console** is displayed.

Once an event has been sent from the **Send Event** dialog, it cannot be canceled or modified in any way.

14.9 Activity Log

The Activity Log process begins when the Production Monitor runs the AutoSys AutoRep command. The AutoRep command reports information about a Job, jobs within boxes, machines, and machine status. A sample Activity Log is illustrated in Figure 14.9-1.

```
        -------------------------------------< Date: 06/14  21:52:04    >-----------------------------
EVENT: CHANGE_STATUS            STATUS:  STARTING        JOB: stage.DPR_04
EVENT: CHANGE_STATUS            STATUS:  RUNNING         JOB: stage.DPR_04
EVENT: CHANGE_STATUS            STATUS:  SUCCESS         JOB: stage.DPR_04

        -------------------------------------< Date: 06/14  21:53:04    >-----------------------------
EVENT: CHANGE_STATUS            STATUS:  STARTING        JOB: prepare.DPR_08
EVENT: CHANGE_STATUS            STATUS:  RUNNING         JOB: prepare.DPR_08
EVENT: CHANGE_STATUS            STATUS:  SUCCESS         JOB: prepare.DPR_08
```

Figure 14.9-1. Sample Activity Log

To display and print the Activity Log use the following procedures:

1 Desktop Access**...**

Access will depend on System Configuration.

2 Enter **/vendor/autotree2/demo/scripts>autorep -J ALL**. (Check on Configuration Path)

- Activity Log will be displayed.
- Enter ***Job_Name*** in place of **ALL** for a specific job.
- Enter **-M *machine_name*** for Machine Report.
- Enter **-s** for summary report.
- Enter **-d** for Detailed Report.
- Enter **-q** for Query Report.

3 Add **| lpr** to the above line to print the document.

- Activity Log will print.

14.10 Job Dependency Log

The Job Dependency Log process begins when the Production Monitor runs the AutoSys Job Depends command. The job_depends command reports information about the dependencies and conditions of a job. This command can be used to determine the current state of a job, its job dependencies, the dependencies and nested hierarchies (for boxes) as specified in the job definition, and a forecast of what jobs will run during a given period of time. A sample Job Dependency Log is illustrated in Figure 14.10-1.

Job Name	Status	Date Cond?	Start Cond?	Dependent Jobs?
DPR##	Activated	No	Yes	No

Condition: (success(DPR_##) and exit code(execute.DPR_##)<5)

Atomic Condition	Current Status	T/F
SUCCESS(SPR_##)	SUCCESS	T
EXIT_CODE(execute.DPR_##)	SUCCESS	F

Figure 14.10-1. Sample Job Dependency Log

To display and print the Job Dependency Log, use the following procedures:

1 Enter UNIX window.

2 Enter **cd /vendor/autotree2/demo/scripts>job_depends -c -J *job_name*** .
- Job Dependency log will be displayed.
- **-c** is used for current condition status.
- **-d** is used for dependencies only.
- **-t** is used for time dependencies.
- **-J *job_name*** indicates the job on which to report. Use **all** for all jobs.

3 Add **| lpr** to the above line to print the document.
- Job Dependency log will print.

14.11 Defining Monitors/Browser

The define monitors/reports process begins when the Production Monitor starts the AutoSys Monitor/Browser from the Production Monitor's desktop. The Monitor/ Browser screen contains fields representing all the information you need to define a monitor or report. See Figure 14.11-1.

```
Alarm: STARTJOBFAIL  Job: execute.DPR_15  06/14  19:18:18  Run #782:9
Exit Code = 0
Job: execute.DPR_15 FAILURE  06/14  19:20:20 Run # 782
<Have EXCEEDED the Max # of times (10) to attempt a restart.  Something is wrong and needs
to be investigated>
Alarm: STARTJOBFAIL  Job: execute.DPR_15  06/14  19:18:18  Run #782:9
Exit Code = -655
```

Figure 14.11-1. Sample Browser Screen

14.11.1 Defining Monitors/Browser

To define Monitors/Reports use the following procedures:

1 Launch AutoSys by **double-clicking** on the **AutoSys** Icon..

_ The **AutoSys** GUI Control Panel is displayed.

2 **Double-click** on the **Monitor/Browser** PushButton.

_ The **Monitor/Browser** GUI page is presented.

3 Enter ***name***, then press **Tab.**

Monitor/Browser defaults are:

Monitor for **Mode**

Alarms for **Types of Events**, **Running**, **Success**, **Failure**

Terminated for **Job Status Events**

ALL Jobs for Job Selection Criteria

Sound for **Monitor Options**.

4 Select **Alarms** or **ALL EVENTS** for Type of Events by **single-clicking** on the ToggleButton:

_ **Light Blue** indicates job status of **success**.

5 Select desired Job Status Events by **single-clicking** on the ToggleButton:

_ Desired Status Events are selected.

6 Select desired Job Selection Criteria by **single-clicking** on the ToggleButton:

_ Either **All Jobs**, **Box with its jobs**, or **Single Job** is selected.

_ If **Single Job** is selected, then enter the ***jobname*** in the **Job Name** field.

7 Select desired **Monitor Option** by **single-clicking** on the ToggleButton:

_ **Sound** or **Verification Required for Alarms** is selected.

8 Select desired Browser Time Criteria by **single-clicking Yes** or **No** for Current Run Time and/or **Events After Date/Time (mm/dd/yy hh:mm)**:

_ Desired Job Selection Criteria is selected.

9 **Single-click Mode** selection of **Monitor** or **Browser**.

_ If **Monitor**, settings are defined for a monitor.

_ If **Browser**, setting are defined for a report.

10 **Single-click Save**.

_ Monitor definition is saved to the database.

You must **Save** the configuration first before results are seen by selecting **Run MonBro**.

11 **Single-click Run MonBro**.

14.11.2 Monitor/Browser Reports

To run a monitor or browser that has been created, use the following procedures:

1 Access the command shell.

_ The Command shell **$** is displayed.

2 Change directory to the AutoSys directory, cd /**TBD - dje**

_ The Command shell **$** is displayed.

3 Report, type **monbro -N *name.***

_ Report is displayed.

Refer to the AutoSys Manual for all options and displays for all **monbro** reports.

The monitor or report (browser) must have be previously defined using the **Monitor/Browser** GUI.

14.12 Database Maintenance Time Change

Once a day, the Event Processor goes into an internal database maintenance cycle. During this time, the Event Processor does not process any events and waits for the maintenance activities to complete before resuming normal operations. The time of day that this maintenance cycle starts up is pre-set to 3:30 PM. If necessary to change the time it runs, you should reset this time to one of minimal activity. The time required for the Database Maintenance cycle is approximately one minute.

To modify the Database Maintenance Time use the following procedures:

1 Access UNIX window.

2 Enter **CD AUTOSYS/install/data/config.ACE**.

_ The configuration file is displayed.

3 Find **DBMaintTime=15:30**. Enter the desired time in 24 hour format.

_ New time is entered.

4 **File** → **Save**.

_ File is saved.

14.13 Production Reports

The Production Reports process begins with the Production Request Editor GUI and opening the Report Generator GUI.

Report options are: Job Dependency Report

Processing Status Report

Processing Errors Report,

Actual Vs Plan Report,

Production Disk Availability Report.

Other reports can be generated from the Monitor/Browser GUI. AutoSys also has the capability to generate Job Reports, Machine Reports, and Reports on AutoSys Global Variables being used.

The following Activity Checklist, Table 14.13-1, provides an overview of generating production reports. 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 14.13-1. Production Reports - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Monitor	Job Dependency Report	(P) 13.7	
2	Production Monitor	Processing Status Report	(P) 13.7	
3	Production Monitor	Processing Errors Report	(P) 13.7	
4	Production Monitor	Actual Vs Plan Report	(P) 13.7	
5	Production Monitor	Production Disk Availability Report	(P) 13.7	
6	Production Monitor	Monitor/Browser Reports	(P) 14.11.2	
7	Production Monitor	AutoSys Reports	(P) 14.13.1	

14.13.1 AutoSys Reports

To print AutoSys reports about: a job, a job within boxes, machines, and machine status, along with reports on global variables being used use the following procedures:

1 Access the command shell.

_ The Command shell **$** is displayed.

2 Change directory to the AutoSys directory, cd /

_ The Command shell **$** is displayed.

3 Detailed Job Report, type **autorep -J *job_name* -d.**

_ Detailed Job Report is displayed.

_ To specify all jobs, enter **all** for ***job_name.***

_ Refer to the AutoSYS Manual for all options and displays for all reports

4 Summary Job Report, type **autorep -J *job_name* -s.**

_ Summary Job Report is displayed.

_ To specify all jobs, enter **all** for ***job_name***

5 Machine Report, type **autorep -M *machine_name.***

_ Machine Report is displayed, including Max Load, Current Load, and Factor.

_ To specify all machines, enter **all** for ***machine_name***

6 AutoSys Global Variables, type **autorep -G.**

_ AutoSys system configuration parameters are displayed.

(**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 14.13-2 presents, in a condensed format, the steps required to print AutoSys reports about: a job, a job within boxes, machines, and machine status, along with reports on global variables being used. 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 the command shell. (**TBD**)
- The Command shell **$** is displayed.

2 Change directory to the AutoSys directory, cd /**TBD**
- The Command shell **$** is displayed.

3 Detailed Job Report, type **autorep -J *job_name* -d.**
- Detailed Job Report is displayed.
- To specify all jobs, enter **all** for ***job_name.***
- See **Note 1.**

4 Summary Job Report, type **autorep -J *job_name* -s.**
- Summary Job Report is displayed.
- To specify all jobs, enter **all** for ***job_name***
- See **Note 1.**

5 Machine Report, type **autorep -M *machine_name.***
- Machine Report is displayed, including Max Load, Current Load, and Factor.
- To specify all machines, enter **all** for ***machine_name***
- See **Note 1.**

6 AutoSys Global Variables, type **autorep -G.**
- AutoSys system configuration parameters are displayed.
- See **Note 1.**

Note 1: Refer to the AutoSys Manual for all options and displays for all reports.

Table 14.13-2. AutoSys Reports - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Access **UNIX command line $**	**Return**
2	Type cd **/TBD**	**Return**
3	Type **autorep -J *job_name* -d.**	**Return**
4	Type **autorep -J *job_name* -s.**	**Return**
5	Type **autorep -M *machine_name.***	**Return**
6	Type **autorep - G.**	**Return**

15. Quality Assurance

The Science Data Specialist performs DAAC Quality Assurance. The Science Data Specialist can input an Operational QA value for a product that is generated through the QA GUI. The operator has the capability to view the product through EOSView and retrieve product history files.

The following Activity Checklist, Table 15.1-1, provides an overview of the quality assurance process. 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 15.1-1. Quality Assurance - Activity Checklist

Order	Role	Task	Section	Complete?
1	Science Data Specialist	Launching the QA Monitor GUI	(P) 15.1	
2	Science Data Specialist	Product QA	(P) 15.2	
3	Science Data Specialist	Retrieve Product History	(P) 15.3	
4	Science Data Specialist	Product Visualization with EOSView	(P) 11.9.1	

15.1 Launching the QA Monitor GUI

In the Pre-Release B testbed, the ECS Desktop is not available. Launching the QA Monitor GUI is a manual operation.

Table 15.1-1 presents the steps required to launch the QA Monitor GUI. 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 ECS System. Enter username and password.

_ Science Data Specialist is able to log in to the ECS System.

2 In the Terminal window, at the command line, enter **cd /usr/ecs/Rel_A/CUSTOM/bin/DPS**.

_ This is the directory containing the Planning Request Editor commands.

3 Set the display environment by entering **setenv DISPLAY ###.###.###.##:0.0** where ###.###.###.## is the IP address of the workstation being used.

_ Display environment is entered

_ See NOTE 1.

4 Enter **DpPrQaMonitorGUI**.

_ The QA Monitor GUI is launched.

15.2 DAAC Product QA

The Product QA process begins with the QA Monitor Application. The Science Data Specialist will Query, Retrieve, and Update (QRU) the selected product. The operator will then retrieve those specific products and perform a visual check of those products using EOSView. Then the operator will update the Operational QA flag for that specific product.

This section explains how to enter DAAC QA results for a Data Product at the Granule level. To enter DAAC Product QA results, execute the procedure steps that follow.

1 Perform procedure 15.1 to invoke the QA GUI.

_ The **Product QA** GUI appears.

_ See **Note 1.**

2 In the **Data Types** panel window, **single-click** the data type to be QA'ed.

a Scroll through the **Data Types** list and single-click the appropriate QA type.
- The Data Type selected will be highlighted.

_ See **Note 2.**

- or -

b Enter the desired Data Type in the Find field and then select the **FIND** button.
- The Data Type selected will be highlighted.

_ See **Note 2.**

3 In the **Data Granule Insert Date** fields, enter mm/dd/yy for Begin and press Tab.

_ MM/DD/YY is displayed for Begin.

_ See **Note 3**.

4 In the **Data Granule Insert Date** fields, enter mm/dd/yy for End.

_ MM/DD/YY is displayed for End.

5 Single-click Query pushbutton.

_ File Names now appear in the Granules window.

6 In the Data Granules: panel window, single-click the data granule to be QA'ed.

a Scroll through the FileName list and single-click the appropriate file.
- The file selected will be highlighted.

_ See **Note 4**.

- or -

b Enter the desired text string in the Find field and then select the **FIND** button.

- The data granule containing the found string will be highlighted.
- Use the Find key to continue searching the list, if necessary.

7 Single-click **Retrieve Data Granule** pushbutton.

_ See **Note 5**.

8 Single-click the **Visualize data** tab.

_ Visualize data screen appears.

9 Filter field contains default configuration directory/file path.

_ Default path is /var/tmp/* showing all files.

_ See **Note 6**.

10 Single-click the file name to be QA'ed in the Files panel window.

_ File name appears in the Selection field.

11 **Single-click** the **Visualize** pushbutton

_ The file is loaded for display. This may take some time depending on file size.

_ See **Note 7**.

12 Review the image checking Operational QA Parameters of the image.

13 **Single-click** the **QRU data** tab.

_ **QRU data** GUI is displayed.

14 **Single-click** the ***File Name*** for the specific in the **Data Granule:** panel window, and **single-click Update MetaFile** pushbutton.

_ **Update MetaFile** window appears.

15 **Single-click** on the **Operational Quality Flag** pushbutton and select desired Operational QA parameter.

_ **N/A, passed**, **failed, being investigated, not being investigated** are QA options.

16 Single-click **Ok**.

_ Operational QA Flag is updated.

17 Select **File - Print** to print QA lists.

_ Window appears with **Print the Data Types List**, **Print the Data Granule List**, and **Print Both**.

18 Select desired print option and select **OK**.

_ QA list is printed.

19 **Select File** → **Quit** when done or repeat specific steps for retrieving, querying, and updating required data granules**.**

Note 1: 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 and then continue on the QA Monitor GUI.

Note 2: The Find field is case sensitive. Only one Date Type can be selected at a time.

Note 3: Time is based upon day of insert into the data server. If no dates are entered an error message will be presented. The up and down arrows to the right of the date fields can be used to advance the date entered in the field.

Note 4: Single-clicking will select a single granule. Pressing the Shift key and single-clicking will select all granules between selections. Pressing the Control key and single-clicking will select multiple individual granules.

Note 5: The retrieve process will take time depending on size and number of files to be retrieved. Have patience.

Note 6: If the operator retrieves one type of files, then retrieves a second type of files, the list of files retrieved will expand. The operator should 'clean out' the /var/tmp directory using UNIX commands as needed to prevent the directory and list of files from becoming too large.

Note 7: Double-clicking on the file name will cause the granule to be visualized.

15.3 Product History

The Product History is retrieved using the QA Monitor GUI. The following is the procedure for performing Retrieve Production history.

1 Perform procedure 15.1 to invoke the QA Monitor GUI.

_ QA Monitor GUI is displayed.

2 Perform procedure 15.2 steps 2 through 6 to select a Data Granule.

_ Data Granule is selected.

3 Single-click **Retrieve Prod History** pushbutton.

_ See **Note 1**.

4 In your Terminal window, enter **cd ./net/sprn2sgi/imf_data/archive** .

_ This is the main directory.

5 In your Terminal window, enter **cd xxxxx** where xxxxx is the Data Type.

_ This is the directory of a specific Data Type.

6 In your Terminal window, enter **more xxxxx.met** where xxxxx is the specific file name that was retrieved.

- This is the metadata for the specific file.
- See **Note 2**.

7 Repeat procedure for other Data Granule Production History files.

Note 1: The retrieve process will take time depending on size of file to be retrieved.

Note 2: The file can be printed using the **ls *filename*.met** command.

15.4 EOSView

The Science Data Specialist will use the EOSView tool to check metadata of a data product. This tool is used specifically during SSI&T "Viewing Product Created Metadata Using the EOSView Tool" procedure in Chapter 11.8.6.

Refer to Chapter 11.8.6 "Viewing Product-Created Metadata Using the EOSView Tool."

This page intentionally left blank.

16. Ingest

This function is not a part of the Pre-Release B Testbed

This page intentionally left blank.