

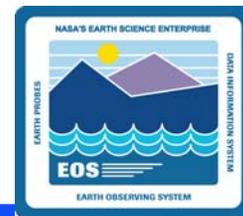
---

# PRODUCTION PLANNING AND PROCESSING

---

**ECS Release 6B Training**

# Overview of Lesson



- **Introduction**
- **Production Planning and Processing (Context)**
- **Production Requests (Including Production Rules)**
- **Logging in to ECS Hosts**
- **Launching the Production Request Editor**
- **Creating/Updating/Deleting a Production Request**
- **Reviewing/Deleting Data Processing Requests**
- **Launching Planning Workbench-Related GUIs**

# Overview of Lesson (Cont.)



- **Creating a New Production Plan (Including Defining a Production Strategy and Reviewing a Plan Timeline)**
- **Cleaning the PDPS Database and Science Processing Disks**
- **Troubleshooting Production Planning Problems**
- **Launching the AutoSys GUI Control Panel**
- **Configuring AutoSys Screens/Displays**
- **Reviewing Hardware Status**
- **Monitoring/Controlling Job Processing**
- **Tuning System Parameters**

# Overview of Lesson (Cont.)



- **Troubleshooting Processing Problems**
- **Launching the Quality Assurance (QA) Monitor**
- **Performing Science Product Quality Assurance**
- **Regenerating Granules in Response to a Loss of Files from the Archive**

# Overview of Lesson (Cont.)



- **Practical Exercise**
  - **Logging in to ECS Hosts**
  - **Launching the Production Request Editor**
  - **Creating a New Production Request Using the Production Request Editor GUI**
  - **Creating a New Production Request Using the Production Request Generator (Command-Line Interface)**
  - **Editing/Modifying a Production Request**
  - **Deleting a Production Request**
  - **Reviewing Data Processing Requests**
  - **Deleting a Data Processing Request**
  - **Launching the Production Strategies GUI**
  - **Launching Production Workbench-Related GUIs**

# Overview of Lesson (Cont.)



- **Practical Exercise (Cont.)**
  - **Defining a Production Strategy**
  - **Creating a New Production Plan**
  - **Reviewing a Plan Timeline**
  - **Cleaning the PDPS Database and Science Processing Disks**
  - **Troubleshooting Production Planning Problems**
  - **Launching the AutoSys GUI Control Panel**
  - **Configuring AutoSys Runtime Options**
  - **Reviewing Hardware Status (Including Changing Hardware Status Views)**
  - **Monitoring/Controlling Job Processing**
  - **Responding to Alarms (Including Selecting Alarms to Be Displayed)**

# Overview of Lesson (Cont.)



- **Practical Exercise (Cont.)**
  - **Specifying Job Selection Criteria and Reviewing Job Activities**
  - **Determining the Ownership of an AutoSys Job**
  - **Modifying Job Status (Including Sending an Event to a Job) Using AutoSys**
  - **Reviewing Activity Reports and Job Dependency Reports**
  - **Defining and Running Monitors/Browsers**
  - **Troubleshooting Processing Problems**
  - **Launching the QA Monitor GUI**
  - **Updating QA Metadata**
  - **Regenerating Granules in Response to a Loss of Files from the Archive**

# Objectives



- **OVERALL:**
  - Develop proficiency in the procedures that apply to production planning and production processing operations
- **SPECIFIC:**
  - Describe the general functions and processes included in the Planning and Data Processing Subsystems (in the context of ECS operations)
  - Perform the steps involved in...
    - » logging in to ECS hosts
    - » launching the production request editor
    - » creating a new production request using the Production Request Editor GUI
    - » creating a new production request using the Production Request Generator (command-line interface)

# Objectives (Cont.)



- **SPECIFIC (Cont.):**
  - **Perform the steps involved in...**
    - » **modifying a production request**
    - » **deleting a production request**
    - » **reviewing data processing requests**
    - » **deleting a data processing request**
    - » **launching the production strategies GUIs**
    - » **launching planning workbench-related GUIs**
    - » **defining a production strategy**
    - » **creating a new production plan**
    - » **reviewing a production plan timeline**
    - » **cleaning the PDPS database and science processing disks**
    - » **troubleshooting production planning problems**

# Objectives (Cont.)



- **SPECIFIC (Cont.):**
  - **Perform the steps involved in...**
    - » **launching the AutoSys GUI Control Panel**
    - » **configuring AutoSys runtime options**
    - » **reviewing hardware status (including changing hardware status views) using AutoSys**
    - » **monitoring/controlling job processing**
    - » **troubleshooting processing problems**
    - » **launching the QA Monitor GUI**
    - » **updating QA metadata**
    - » **regenerating granules in response to a loss of files from the archive**

# Objectives (Cont.)



- **STANDARDS:**
  - Lesson content (procedures in the lesson)
  - Mission Operation Procedures for the ECS Project (611-CD-610-002)

# Production Planning and Processing



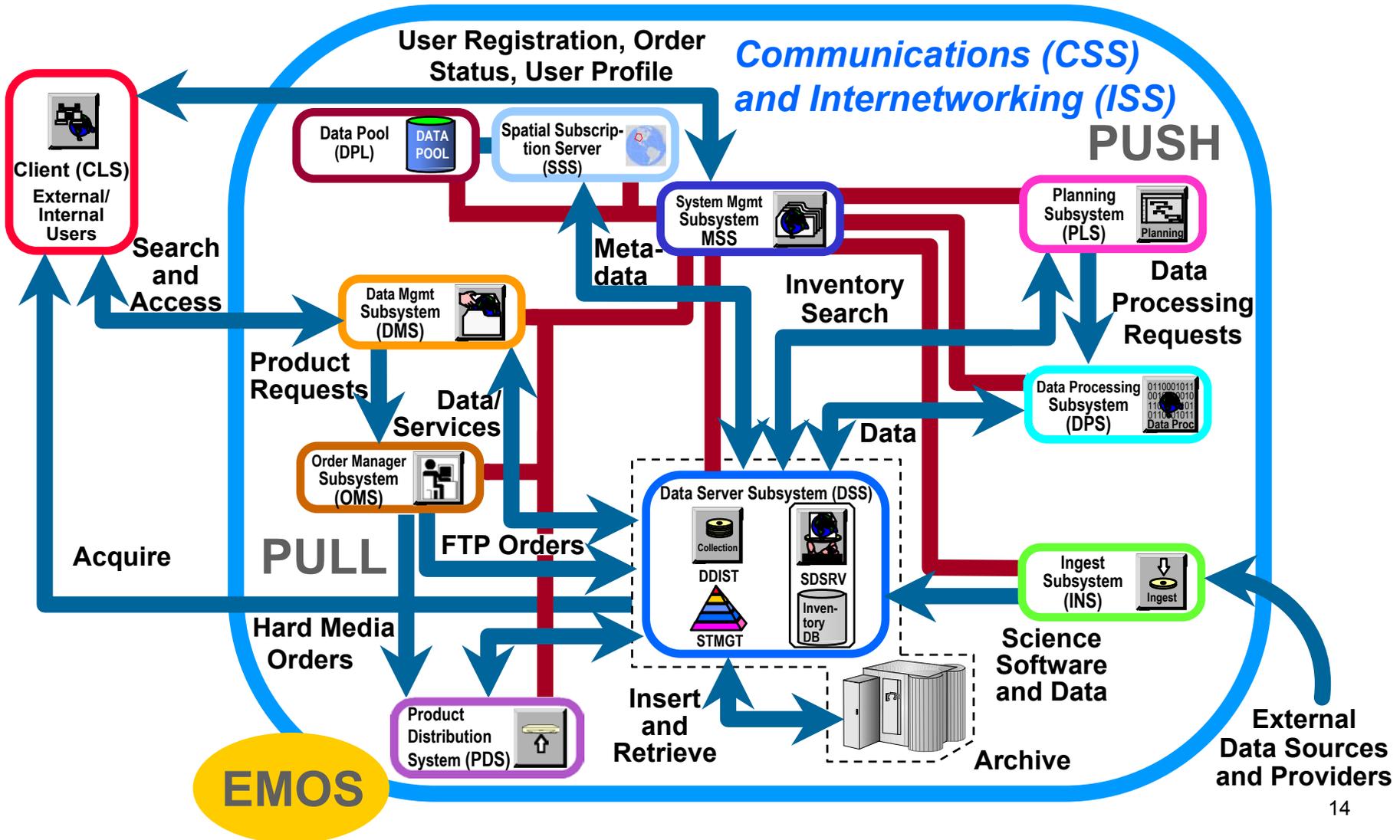
- **ECS Context**
  - **ECS production planning and processing processes are accomplished at the Distributed Active Archive Centers (DAACs)**
  - **People involved in production planning and processing activities are...**
    - » **Production Planners**
    - » **Production Monitors**

# Production Planning and Processing (Cont.)



- **ECS Context (Cont.)**
  - **Production Planner**
    - » performs planning functions; especially, using the **Planning Subsystem (PLS)** to create **Data Processing Requests** and specify which requests are to be processed as part of a particular **Production Plan**
  - **Production Monitor**
    - » keeps track of operations in the **Data Processing Subsystem**, especially the execution of science data processing jobs (creation of data products)

# ECS Context Diagram



# Production Planning and Processing (Cont.)

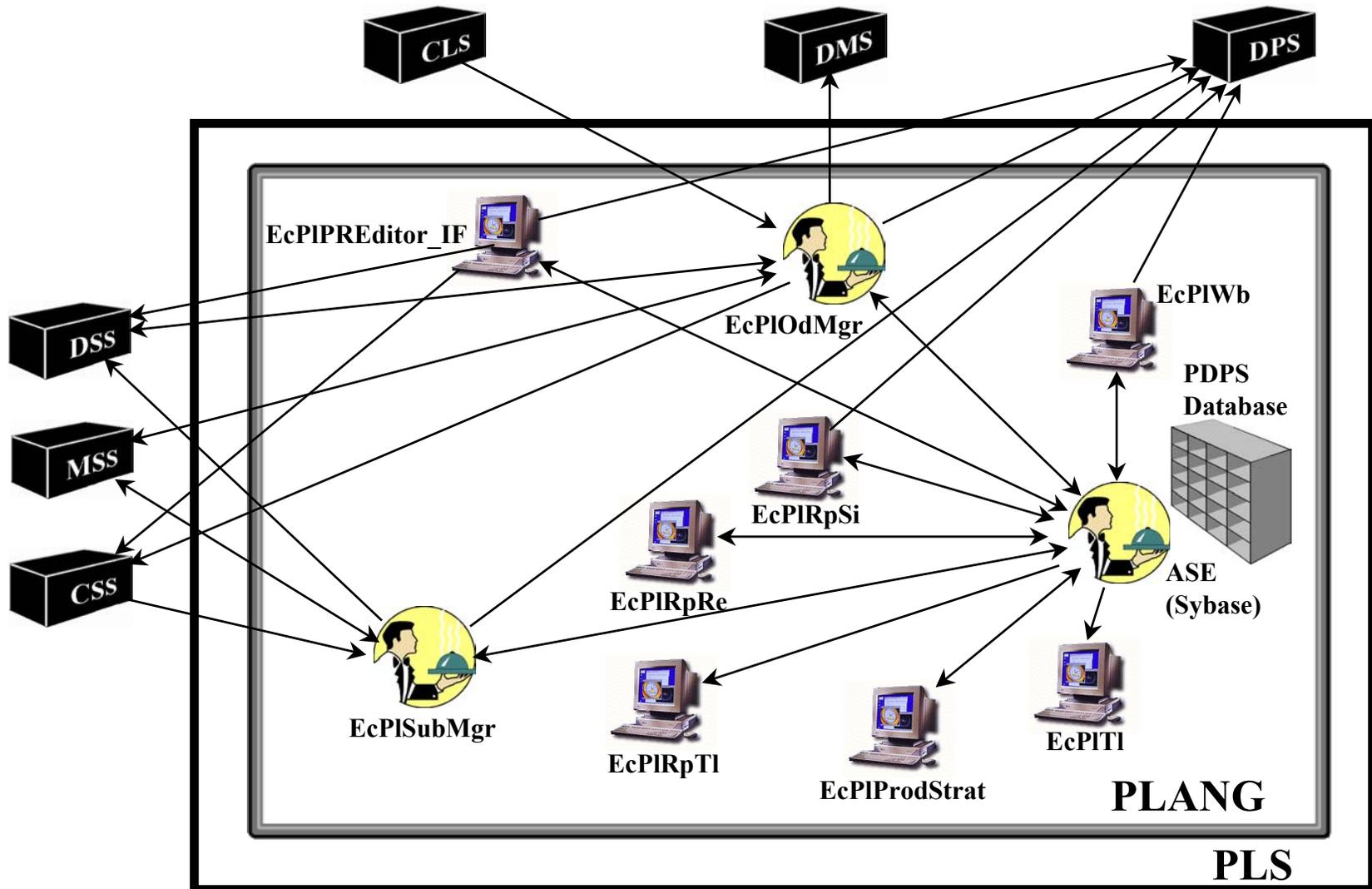


- **Planning Subsystem**

- **provides a mechanism for accomplishing the following general functions:**

- » **Defining DAAC production resources**
    - » **Scheduling production resources for non-production-related activities**
    - » **Defining data processing jobs to be performed at the DAAC**
    - » **Generating efficient plans for scheduling defined data processing jobs**
    - » **Coordinating production with the Data Server Subsystem and Data Processing Subsystem to achieve a highly automated production system**

# Planning Architecture

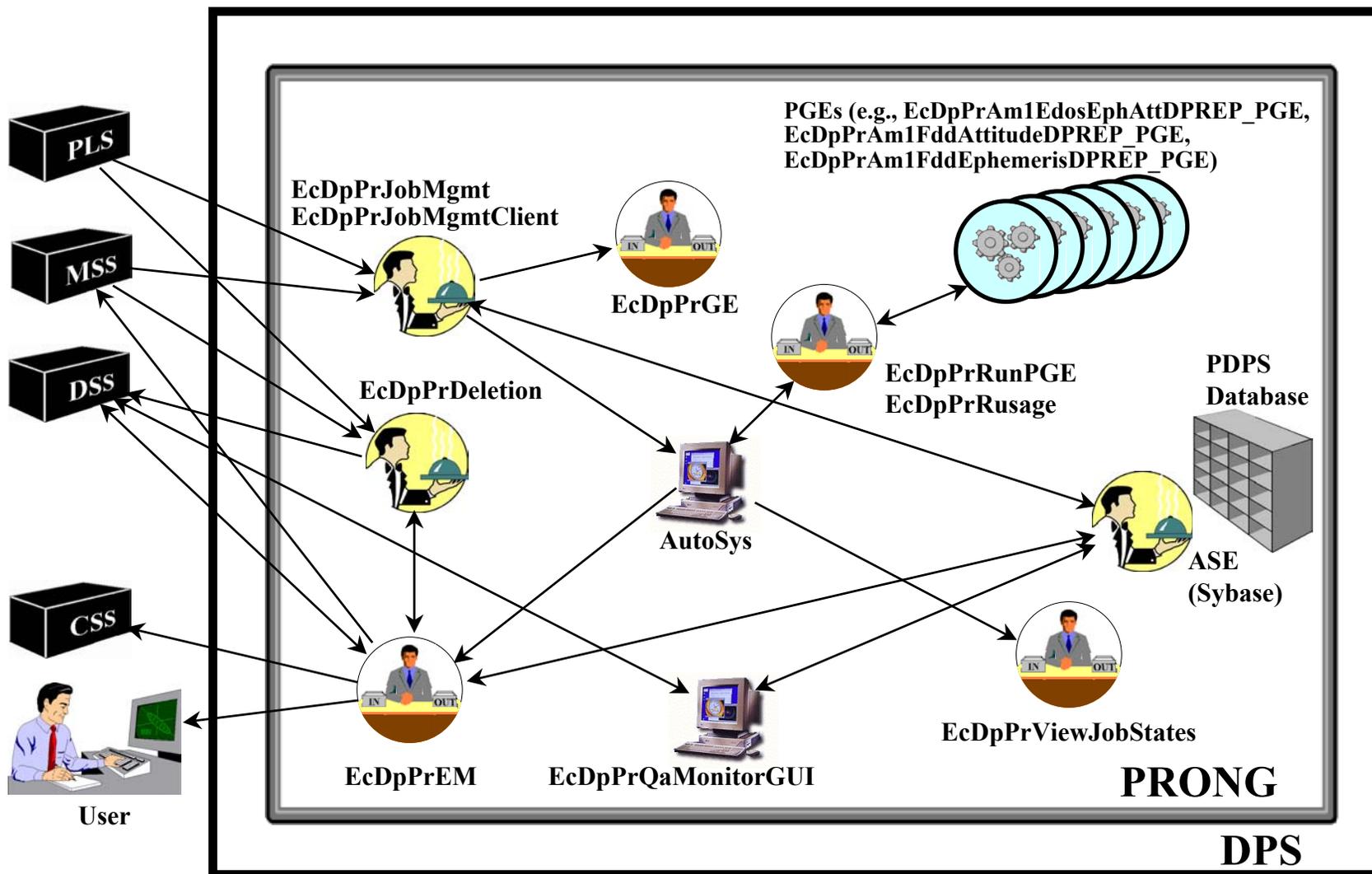
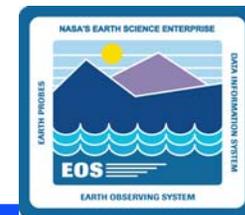


# Production Planning and Processing (Cont.)



- **Data Processing Subsystem**
  - provides a mechanism for accomplishing the following general functions:
    - » Managing the allocation of data processing jobs to the site's data processing resources
    - » Managing, queuing, and executing data processing jobs to produce data products
    - » Supporting preliminary processing of ancillary data granules

# Data Processing Architecture (PRONG)



# Production Planning and Processing (Cont.)



- **PLANG is the Planning Subsystem computer software configuration item (CSCI)**
  - **Resource Planning Workbench**
    - » **Resource Editor (EcPIRpRe)**
    - » **Resource Scheduler (EcPIRpSi)**
    - » **Resource Reservation Planning Master Timeline GUI (EcPIRpTI)**
  - **Production Request Editor (EcPIPREditor)**
  - **Production Planning Workbench**
    - » **Planning Workbench GUI (EcPIWb)**
    - » **Production Strategies GUI (EcPIProdStrat)**
    - » **Planning Master Timeline GUI (EcPITI)**

# Production Planning and Processing (Cont.)



- **PLANG (Cont.)**
  - **On-Demand Manager (EcPIOdMgr)**
  - **On-Demand Manager Client (EcPIOdMgrClient)**
  - **Subscription Manager (EcPISubMgr)**
  - **Sybase Adaptive Server Enterprise (ASE) Server**
  - **Message Handler (EcPIMsh)**
  - **System Name Server (EcPISns)**
  - **Resource Model (EcPIRpRm, EcPIRm)**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Start-up and shutdown scripts used by planning personnel (/usr/ecs/MODE/CUSTOM/utilities directory on the Planning/Management Workstation)**
    - » **EcPISomeStart**
    - » **EcPIAllStart**
    - » **EcPIPRE\_IFStart**
    - » **EcPIPRE\_ReadOnlyStart**
    - » **EcPIPRGeneratorStart**
    - » **EcPIProdStratStart**
    - » **EcPIRpAllStart**
    - » **EcPIRpReStart**
    - » **EcPIRpSiStart**
    - » **EcPISubsEditStart**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Start-up and shutdown scripts used by planning personnel (/usr/ecs/MODE/CUSTOM/utilities directory on the Planning/Management Workstation) (Cont.)**
    - » **EcPITStart**
    - » **EcPIWbStart**
    - » **EcPISlay**
    - » **EcPISlayAll**
    - » **EcPIRpSlayAll**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Start-up scripts in the /usr/ecs/MODE/CUSTOM/utilities directory on the Queuing Server:**
    - » **EcPIOdMgrClientStart**
    - » **EcPIOdMgrStart**
    - » **EcPIPlanningAppStart**
    - » **EcPIStart**
    - » **EcPISubMgrStart**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Start-up scripts called by other applications (not normally invoked directly by planning personnel)**
    - » **EcPIMshStart**
    - » **EcPIRmStart**
    - » **EcPIRpRmStart**
    - » **EcPISnsStart**
    - » **EcPIStart**
    - » **SweeperStart**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Other scripts**
    - » **EcLgLogCtrlStart**
    - » **EcPICdsPingServers**
    - » **EcCsPerfLogProcessor.pl**
    - » **EcPIDbClean**
    - » **EcPIDbCleanArchive**
    - » **EcPIDbBuild**
    - » **EcPIDbDrop**
    - » **EcPIDbDump**
    - » **EcPIDbMigrate**
    - » **EcPIDbPatch**
    - » **EcPIDetermineChain.pl**
    - » **EcPIRpFetchBaseline**

# Production Planning Concepts (Cont.)



- **PLANG (Cont.)**
  - **Other scripts (Cont.)**
    - » **EcPIDbReset**
    - » **EcPIDbList**
    - » **EcPIDbSave**
    - » **fos\_services**

# Production Planning and Processing (Cont.)



- **Data Processing Subsystem is composed of three computer software configuration items (CSCIs):**
  - **PRONG**
    - » Provides the services required to manage and monitor the Science Data Processing environment, which executes Science Software items (PGEs) and produces data products
  - **Algorithm Integration & Test Tools (AITTL)**
    - » Set of tools used for test and integration of new science software, new versions of science software, and user methods into the Science Data Processing operational environment
  - **Science Data Processing (SDP) Toolkit**
    - » Provides a set of software libraries which are used to integrate Science Software into the ECS environment

# Production Planning and Processing (Cont.)



- **PRONG CSCI**
  - **Job Management (EcDpPrJobMgmt)**
  - **Ground Event process (EcDpPrGE)**
  - **Job Management Client (EcDpPrJobMgmtClient)**
  - **AutoSys/AutoXpert**
    - » **Event Processor (AutoSys daemon)**
    - » **Event Server**
    - » **AutoSys GUIs**
    - » **AutoXpert GUIs**
  - **Execution Management (EcDpPrEM)**
  - **PGE Execution Manager (EcDpPrRunPGE)**
  - **Resource Usage (EcDpPrRusage)**

# Production Planning and Processing (Cont.)



- **PRONG CSCI (Cont.)**
  - **View Job States (EcDpPrViewJobStates)**
  - **Terra Data Preprocessing (DPREP)**
    - » **AM1Eph or Step 1**  
**(EcDpPrAm1EdosEphAttDPREP\_PGE)**
    - » **FddAtt or Step 2**  
**(EcDpPrAm1FddAttitudeDPREP\_PGE)**
    - » **RepEph or Step 3**  
**(EcDpPrAm1FddEphemerisDPREP\_PGE)**
    - » **EcDpPrDumpAttitudeDPREP**
    - » **EcDpPrDumpEphemerisDPREP**

# Production Planning and Processing (Cont.)



- **PRONG CSCI (Cont.)**
  - **Aqua DPREP**
    - » **PM1DefEph (EcDpPrPm1FddEphemerisDPREP\_PGE)**
    - » **PM1DefAtt (EcDpPrPm1AttitudeDPREP\_PGE)**
  - **Aura DPREP**
    - » **AuraEph (EcDpPrAuraEphemerisDPREP\_PGE)**
    - » **AuraAtt (EcDpPrAuraAttitudeDPREP\_PGE)**
  - **Deletion Server (EcDpPrDeletion)**
  - **Deletion Client (EcDpPrDeletionClient)**
  - **Sybase ASE Server**
  - **Quality Assurance Monitor (EcDpPrQaMonitorGUI)**
  - **EcDpPrLoadTable.pl**
  - **EcDpPrAutocons**

# Production Planning and Processing (Cont.)



- **PRONG CSCI (Cont.)**
  - **EcDpPrEMGetAncHeaders**
  - **EcDpPrPREPQCConverterPGE**
  - **EcDpPrReadPREPQCData**
  - **EcDpPrWritePREPQCDataToHDFEOS**
  - **EcDpPrSMFCopy**

# Production Planning Concepts (Cont.)



- **PRONG (Cont.)**
  - **Start-up scripts used by production personnel (/usr/ecs/MODE/CUSTOM/utilities directory on the Queuing Server)**
    - » **EcDpPrAutosysStart**
    - » **EcDpPrDeletionClientStart**
    - » **EcDpPrGarbageCollectorStart**
  - **Start-up scripts called by other applications (not normally invoked directly by production personnel)**
    - » **EcDpPrDeletionStart**
    - » **EcDpPrDisplayJobStates**
    - » **EcDpPrJobMgmtClientStart**
    - » **EcDpPrJobMgmtStart**
    - » **EcDpPrStart**
    - » **EcDpProcessingAppStart**

# Production Planning Concepts (Cont.)



- **PRONG (Cont.)**
  - **Start-up script used by production personnel (/usr/ecs/MODE/CUSTOM/utilities directory on the Planning/Management Workstation)**
    - » **EcDpPrQaMonitorGUIStart**
  - **Other scripts available on the Queuing Server:**
    - » **EcDpBusySystemClean.pl**
    - » **EcDpPrRestartFailedJobs**
    - » **EcDpPrCleanMaintMachineFiles.pl**
    - » **EcDpPrRmFilesWOGranules.pl**

# Production Requests



- **Production planning at the Distributed Active Archive Centers (DAACs)**
  - process by which the Production Planner notifies the Planning Subsystem (PLS) of the science processing jobs that are to be processed and what data to process
- **Science Software (SS)**
  - does the actual data processing
  - is developed at Science Computing Facilities (SCFs)
  - is embodied in Product Generation Executives (PGEs) when the software is integrated into the ECS production processing environment



# Production Requests (Cont.)

- **PGEs**
  - science software code (e.g., executable programs or shell scripts) that contain the instructions for processing data to create the desired products
- **Production Request (PR)**
  - Production Planner defines ECS science data processing jobs in terms of PRs
  - A PR is an order for data to be produced by the data processing system
  - A single PR may specify...
    - » several jobs that are to be run over a period of time
    - » a single job producing a single set of data



# Production Requests (Cont.)

- **Production Request (Cont.)**
  - may apply to the processing of new data (standard PRs or standing orders)
  - may apply to the reprocessing of existing data (reprocessing PRs)
  - each PR identifies a specific PGE for generating a particular type of product
    - » some PGEs are dependent on others; i.e., some PGEs require input data that are the output of other PGEs
  - planning software will recognize and reject a PR when the PR specifies a PGE that requires data from another PGE that has not yet been specified in a PR

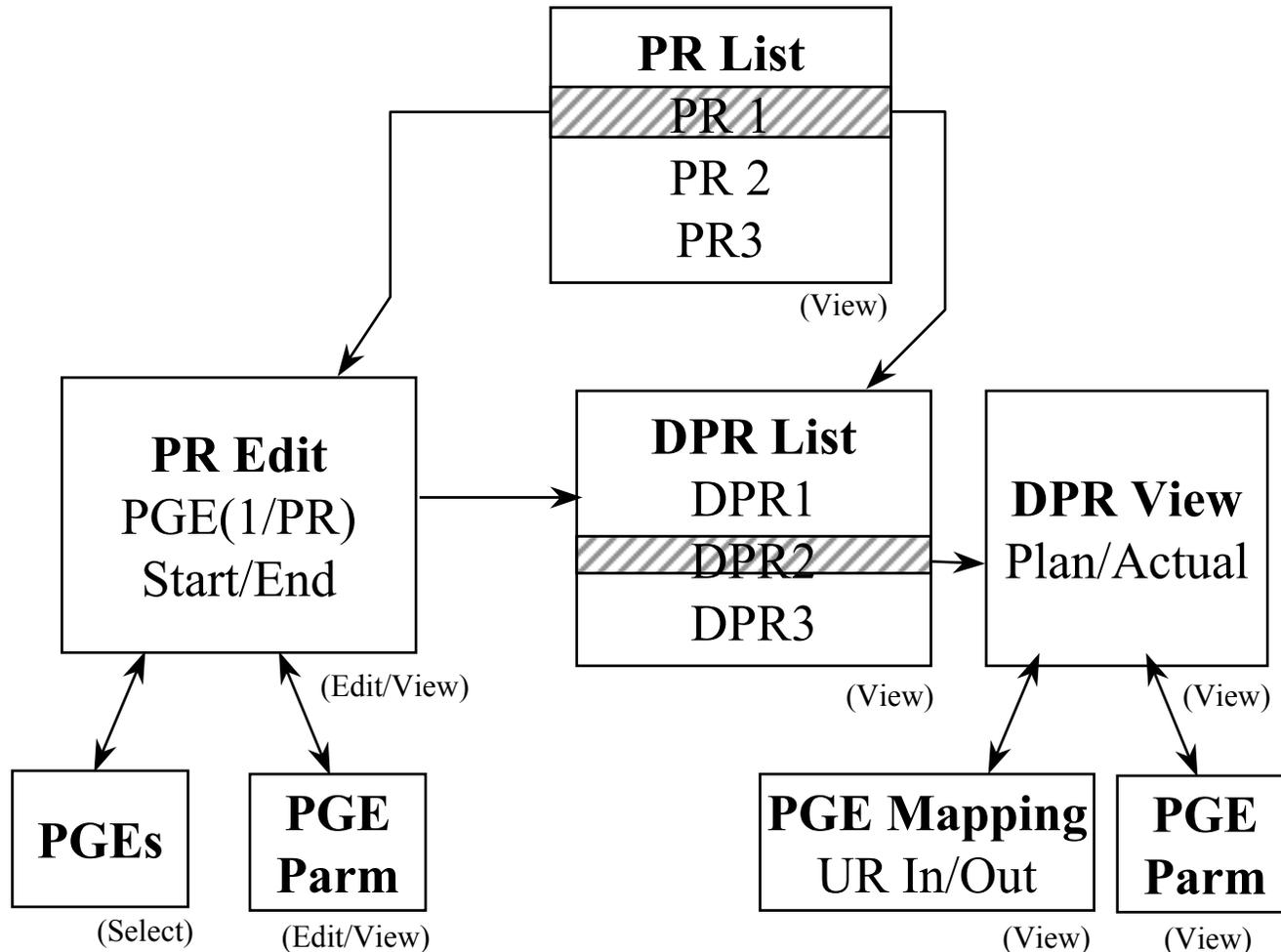
# Production Requests (Cont.)

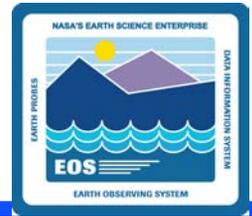


- **Planning Subsystem**

- uses each PR to generate either one or a series of **Data Processing Requests (DPRs)**
  - » each DPR corresponds to one execution of a single **PGE**
  - » each DPR contains the information that is needed by the **SDPS processing function**
- checks the availability of the data required for the **DPR**
- determines what data will be included in the **DPR output**
  - » so the system can make predictions concerning the **future availability of data**

# Production Requests (Cont.)





# Production Requests (Cont.)

- **Types of Processing**
  - **Routine Processing**
    - » pre-defined software production processing that is periodic and keyed to data arrival
  - **Reprocessing**
    - » using a new, improved PGE to process data that had previously been processed with an older version of the PGE
  - **Regeneration**
    - » a type of reprocessing performed for the purpose of replacing a missing or damaged product
  - **On-Demand Processing**
    - » ad-hoc processing initiated by an end-user (as opposed to the Production Planner)

# Production Requests (Cont.)



- **SCF Processing Requests**

- **Reference:**

- » **Interface Control Document Between EOSDIS Core System (ECS) and Science Computing Facilities (SCF) [505-41-33]**

- **SCFs may request the DAACs to process or reprocess data produced using SCF-provided science software**
  - **SCFs' processing/reprocessing request messages can include optional run-time parameters, in case of changes from the original processing parameters**

# Production Requests (Cont.)



- **SCF Processing Requests (Cont.)**
  - **SCF staff send Processing Requests to the DAAC staff using one of the following three protocols:**
    - » e-mail
    - » X11/internet
    - » X11/modem
  - **SCFs must use the e-mail interface unless the DAAC has authorized their use of the X11 Interfaces**
    - » **Specific protocol used in each instance will have been defined in an Operations Agreement (or lower-level type of document) between SCF and DAAC**



# Production Requests (Cont.)

- **SCF Processing Requests (Cont.)**
  - **SCF Processing Requests provide DAAC operations staff with the following information:**
    - » **PGE Name**
    - » **PGE Version**
    - » **Optional PGE Profile (number 1-99 defining how the PGE is to be run)**
    - » **Time window for which the product should be regenerated**
    - » **Optional run-time parameters**
    - » **Optional comment**
  - **DAAC operations personnel acknowledge receipt of an SCF's Processing Request by replying to the SCF in an e-mail message that includes the SCF's original request and the following statement:**
    - » **"This is an acknowledgment of the receipt of the following Processing Request: . . . "**

# Production Requests (Cont.)



- **Production Rules**
  - Instructions about how a particular PGE is to be run
  - Specify a wide range of information such as...
    - » input and output data types
    - » frequency of execution
    - » activation conditions
    - » error handling instructions
  - PGE profiles
    - » permit a PGE to use different sets of production rules
    - » allow running a PGE with different input data sets, or activation conditions



# Production Requests (Cont.)

- **Production Rules (Cont.)**
  - Are entered when a PGE undergoes Science Software Integration and Test (SSI&T) at the DAAC
  - Where applicable, default parameter values are entered during SSI&T
  - Some parameters can be overridden when a Production Request is entered
  - Types of conditions that can be specified for each PGE:
    - » time period for which the PGE will run
    - » PGE inputs
    - » PGE outputs
    - » runtime parameter values
  - Some (but not all) production rules can work with other production rules

# Production Requests (Cont.)



- **Production Rules (Cont.)**

- **Production rules are often used for the selection of dynamic inputs**

- » **dynamic internal inputs are produced by other PGEs (they are called dynamic internal inputs because they are produced at an ECS DAAC)**
    - » **dynamic external inputs are periodically ingested and stored in the Data Server Subsystem (they are termed dynamic external inputs because they are produced outside of the DAAC)**
    - » **static inputs are granules that are inserted during the SSI&T process and are retrieved not on the basis of time but by Earth Science Data Type (ESDT) and science group**



# Production Requests (Cont.)

- **Production Rules (Cont.)**
  - **Methods of defining production rules**
    - » **science metadata is entered in various types of files during the SSI&T process**
    - » **parameter values are entered when a Production Request is created to schedule the PGE**
  - **During SSI&T at the DAAC information concerning the production rule(s) applicable to the PGE is included in Object Description Language (ODL) files**
  - **ODL files use 'parameter equals value' format**

# Production Requests (Cont.)



- **Production Rules (Cont.) - Categories of ODL files**
  - **PGE Science Metadata ODL Files**
  - **ESDT Science Metadata ODL Files**
  - **Production Rule-Specific Science Metadata ODL Files**
    - » **Orbit Definition ODL Files**
    - » **Path Map Definition ODL Files**
    - » **Tile Science Metadata ODL Files**



# Production Requests (Cont.)

- **Production Rules**
  - **Basic Temporal**
    - » Temporal (time) range of inputs matches the temporal range of outputs
  - **Advanced Temporal**
    - » Temporal range of inputs is offset from the expected temporal range of inputs and outputs
  - **Alternate Inputs**
    - » PGE is run with different inputs based on the availability or quality of various alternate input data sets
  - **Optional Inputs**
    - » PGE is run with specified optional inputs if available; otherwise, PGE is run without them



# Production Requests (Cont.)

- **Production Rules (Cont.)**
  - **Minimum/Maximum Number of Granules**
    - » **Minimum number of input granules needed for full data coverage and maximum number of input granules to search for may be specified**
    - » **Minimum and maximum number of outputs expected from the PGE may be specified**
  - **Optional DPRs**
    - » **The only DPRs executed are those for which the non-routine key input data actually become available (i.e., are either produced in data processing or can be acquired from the archive)**

# Production Requests (Cont.)



- **Production Rules (Cont.)**

- **Metadata Checks**

- » DPR is run only if input data's metadata value(s) meet(s) certain criteria

- **Metadata Query**

- » Input granule selection is based on metadata value

- **Spatial Query/Spatial Pad**

- » Input granule selection is based on the spatial coverage of another input (i.e., the key input)
    - » Spatial Pad involves adding area to all sides of the key input's spatial shape



# Production Requests (Cont.)

- **Production Rules (Cont.)**
  - **Closest Granule**
    - » DPR is generated if a required input granule within a particular time range (rather than an exact time) is available; otherwise, no DPR is generated
    - » Supersedes the Most Recent Granule Production Rule
  - **Orbital Processing**
    - » Selection of input times is based on orbit information
  - **Multiple DPRs for Insertion Time**
    - » Allows the creation of DPRs for multiple granules with the same insertion time (affects ASTER L1B routine processing only)

# Production Requests (Cont.)



- **Production Rules (Cont.)**
  - **Tiling**
    - » **Input data is chosen on the basis of Instrument Team-defined tiles (geographic areas)**

# Production Requests (Cont.)



- **Basic Temporal Production Rule**
  - Defines the time frame for the PGE along with its input and output data
  - Typically scheduled to run using input data that become available periodically (every hour, every day, etc.)
  - Use input data for a particular period of time
  - Produce output for a specified length of time
  - Data the PGE takes in (its input) and the data it produces (its output) have the same period (or some subset of the same period) as the PGE



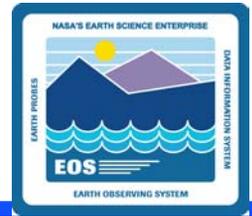
# Production Requests (Cont.)

- **Basic Temporal Production Rule (Cont.)**
  - **Example One:**
    - » **A MODIS PGE processes data for five-minute intervals, producing Level 1B granules**
    - » **The PGE requires as input the specific five-minute Level 1A granule that is contemporaneous with (covers the same five-minute time period as) the Level 1B granule to be produced**
    - » **Using the Basic Temporal Production Rule, a five-minute Level 1A granule is staged as input to the PGE and a five-minute Level 1B granule is expected as output, both matching the time frame for which the PGE is run**

# Production Requests (Cont.)



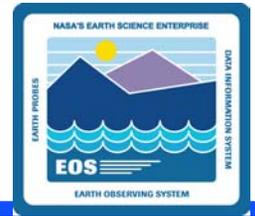
- **Basic Temporal Production Rule (Cont.)**
  - **Example Two:**
    - » **A PGE for a different instrument processes data for 24-hour intervals, producing 24-hour Level 1A granules as output**
    - » **As input the PGE takes Level 0 data that is ingested every two hours**
    - » **Using the Basic Temporal Production Rule, twelve two-hour Level 0 granules are staged as input to the PGE and a 24-hour Level 1A granule is expected as output, matching the time frame for which the PGE is run**



# Production Requests (Cont.)

- **Basic Temporal Production Rule (Cont.)**
  - **Fundamental elements used to define the Basic Temporal Production Rule are...**
    - » **period**
    - » **boundary**
  - **Period is the length of time for which a PGE processes data or the length of time for which input and output data is collected**
    - » **A PGE that is subject to the Basic Temporal Production Rule only and that processes data in two-hour blocks, takes in data that relates to a particular two-hour interval and produces output data for that same two-hour period**
    - » **Data that has a period of 15 minutes was collected or produced for a 15-minute time period**

# Production Requests (Cont.)



- **Basic Temporal Production Rule (Cont.)**
  - **Boundary is the starting point for the data or PGE**
    - » Depending on the characteristics of the data or PGE, the boundary may be the start of a minute or hour or day or week (etc.)
    - » If a PGE's boundary is the start of the hour, it processes data that starts every hour and runs on data for the length of its period
    - » If data comes in every day, PDPS predicts that the data is going to be available at the start of the day and allows scheduling of PGEs that use the data as input accordingly



# Production Requests (Cont.)

- **Basic Temporal Production Rule (Cont.)**
  - Both the PGE itself and the input data have a boundary and period associated with them
    - » So PDPS can determine the frequency of processing for a Basic Temporal PGE and the time period for its inputs and outputs
  - PDPS uses period and boundary in combination to plan the processing of each PGE, including determining its input requirements and anticipated output (which may be input to other PGEs)
    - » If a PGE has a period of one hour and a boundary of “start of day,” it is scheduled every hour, beginning at midnight
    - » If an input has a period of 15 minutes and boundary of “start of hour,” PDPS predicts it every 15 minutes beginning on the hour



# Production Requests (Cont.)

- **Basic Temporal Production Rule (Cont.)**
  - **Additions to the Basic Temporal Production Rule**
    - » **boundary offset**
    - » **end-of-month anomaly**
  - **Illustration of the Basic Temporal Production Rule**
    - » **PGE boundary is “start of day”**
    - » **PGE period is one hour**
    - » **PGE is scheduled for every hour through the day**
    - » **If a Production Request were entered for two full days of processing, a DPR would be created for the PGE to run every hour; i.e., 48 DPRs total**
    - » **If a Production Request were created for a four-hour period in the middle of a single day (for example, from 12:00 noon to 4:00 p.m.), then four DPRs would be created**

# Example of Basic Temporal Production Rule

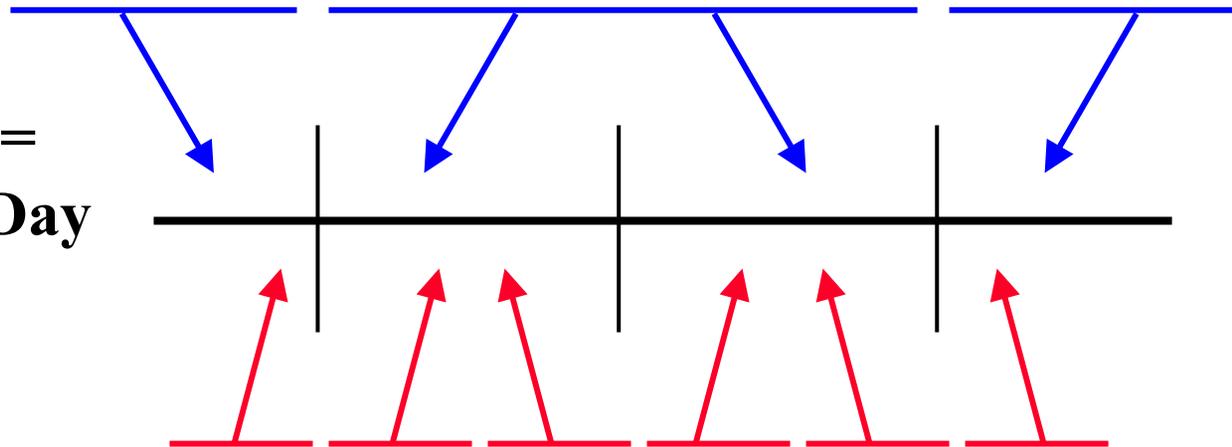


**Input One: Boundary = Start of Day  
Period = 2 hours**

**PGE**

**Boundary =  
Start of Day**

**Period =  
1 hour**



**Input Two: Boundary = Start of Day  
Period = 1/2 hour**



# Production Requests (Cont.)

- **Advanced Temporal Production Rule**
  - **Allows for input data to be acquired for a time period other than that of the PGE or its planned inputs/outputs**
  - **Offset mechanism**
    - » **specifies on an input basis that the data required for processing is some number of seconds earlier or later than the planned time period for the PGE**
  - **Example One:**
    - » **A PGE requires data from its previous execution for interpolation purposes (e.g., one of its inputs is the output of the very same PGE the last time that it ran)**
    - » **If the PGE processes data for each one-hour interval (producing an hourly product), the Advanced Temporal Production Rule is specified with an offset of minus 3600 seconds (one hour) for the input of the ESDT produced by previous runs**

# Production Requests (Cont.)



- **Advanced Temporal Production Rule (Cont.)**
  - **Example Two:**
    - » **A PGE takes as input two-hour Level 0 data to produce an L1A product**
    - » **Because the edges of the Level 0 data can be difficult to process without preceding and succeeding data, the PGE requires three Level 0 granules, one from the time period before it runs, one for the time period it is currently processing and one for the next time period**
    - » **The PGE is defined as having three inputs, the first with an Advanced Temporal offset of minus 7200 seconds (two hours), the second with no Advanced Temporal offset and the third with an Advanced Temporal offset of plus 7200 seconds (two hours)**

# Production Requests (Cont.)



- **Advanced Temporal Production Rule (Cont.)**
  - Uses the times specified in the Basic Temporal Production Rule as a reference point for specifying offset(s) to request data from a “period” and/or “boundary” different from that of the DPR or its input
  - **Offsets ...**
    - » are specified as either negative or positive numbers
    - » indicate whether the time period of the input data is before or after that of the DPR (a particular run of a PGE)



# Production Requests (Cont.)

- **Advanced Temporal Production Rule (Cont.)**
  - **Begin Period Offset**
    - » an amount of time (in seconds) that is specified with respect to the DPR start time
    - » negative beginning offset requests data that was collected before the DPR start time
    - » positive beginning offset requests data with a collection time after the start time of the DPR
  - **End Period Offset**
    - » an amount of time (in seconds) that is specified with respect to the DPR end time
    - » negative ending offset requests data that ended collection before the DPR end time was reached
    - » positive ending offset requests data that ended collection after the end time of the DPR boundaries
  - **Beginning and ending offsets are not absolute cut-offs for data (overlapping granules are staged)**

# Production Requests (Cont.)



- **Advanced Temporal Production Rule (Cont.)**
  - **Illustration of the Advanced Temporal Production Rule**
    - » **PGE processes data for every one-hour interval**
    - » **Input One comes in at two-hour intervals**
    - » **Input Two is produced every 1/2 hour**
    - » **Input One: both the Begin Period Offset and End Period Offset are -7200 seconds (minus two hours)**
    - » **Every DPR will stage the "previous" Input One (this could be used to get the "previous" or "next" granule of an input)**

# Production Requests (Cont.)



- **Advanced Temporal Production Rule (Cont.)**
  - **Illustration of the Advanced Temporal Production Rule (Cont.)**
    - » **Input Two Begin Period Offset for is zero (it will match the Start Time of the DPR)**
    - » **Input Two End Period Offset is +1800 seconds (plus 1/2 hour)**
    - » **All Input Two granules that fall within the time period of the DPR plus 1/2 hour would be staged**
    - » **All Input Two granules within the time period of the DPR are acquired, plus the one from the next 1/2-hour time period, for a total of three granules**
    - » **The additional granule acquired by means of the End Period Offset might be used for interpolation purposes at the end point**

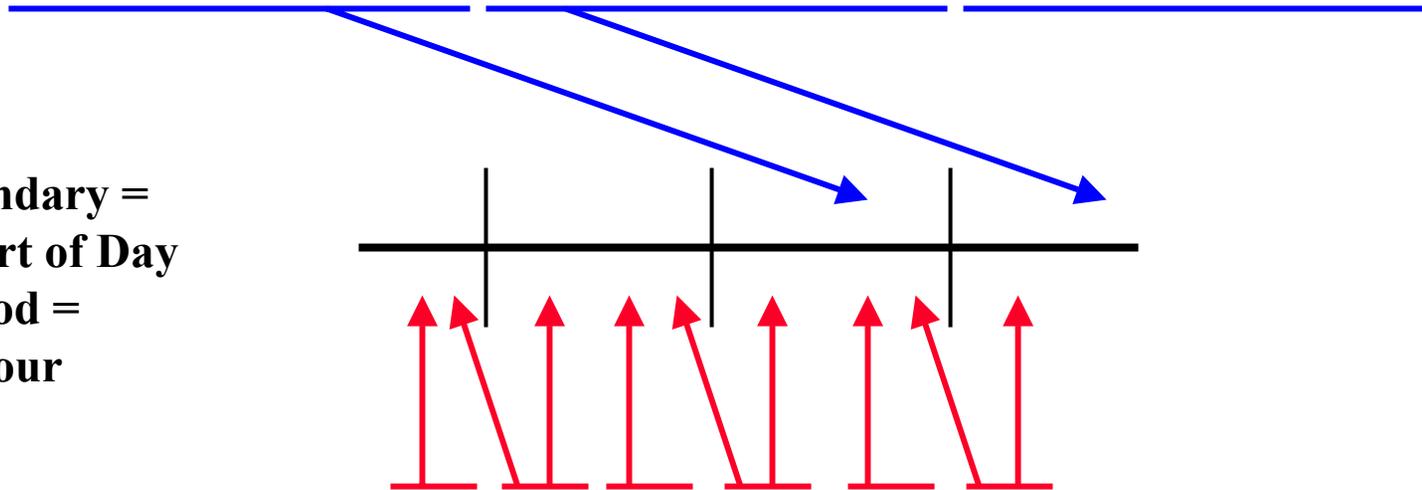
# Example of the Advanced Temporal Production Rule



**Input One: Boundary = Start of Day**  
**Period = 2 hours**  
**Begin Period Offset = -7200 (-2 hours)**  
**End Period Offset = -7200 (-2 hours)**

**PGE**

**Boundary =**  
**Start of Day**  
**Period =**  
**1 hour**



**Input Two: Boundary = Start of Day**  
**Period = 30 minutes (1/2 hour)**  
**Begin Period Offset = 0**  
**End Period Offset = +1800 (+1/2 hour)**

# Production Requests (Cont.)



- **Alternate Inputs and Optional Inputs Production Rules**
  - **Very similar rules that involve much the same processing in PDPS**
  - **Allow a PGE to select various inputs based on timers and priority lists**
  - **Major difference:**
    - » **Alternate Inputs requires that one of alternates on the list be used**
    - » **Optional Inputs allows successful execution of the PGE if no optional input on the list is available**

# Production Requests (Cont.)



- **Alternate Input Production Rule**
  - A list of inputs for a PGE is evaluated in priority order
    - » PGE is scheduled and executed with the best priority input that could be found
  - A timer can be used to specify how long to wait for a given alternate choice before proceeding with a choice of lesser priority
  - PGE is not executed until one of the alternate choices has been found



# Production Requests (Cont.)

- **Alternate Input Production Rule (Cont.)**

- **Example:**

- » **PGE requires model wind data as an input but is capable of accepting wind data from a Data Assimilation Office (DAO) model, a National Centers for Environmental Prediction (NCEP) model, or climatology**
    - » **Each input is considered in priority order**
    - » **A timer value indicates how long to wait before trying the next input**
    - » **DAO is listed as first choice or "primary" data**
    - » **NCEP is the second choice**
    - » **Climatology is the last choice**
    - » **PGE waits for the DAO timer to expire before running with either NCEP data or climatology**
    - » **PGE waits for the NCEP timer to expire before running with the climatology data**

# Production Requests (Cont.)



- **Optional Input Production Rule**
  - There is a list of inputs that are desired but not required for execution of a PGE
  - Inputs are ranked in priority order
  - Timers are set to wait before choosing a lower-priority type of input
  - If none of the inputs on the list becomes available, the PGE starts because the alternatives are classified as "optional"

# Production Requests (Cont.)



- **Optional Input Production Rule (Cont.)**

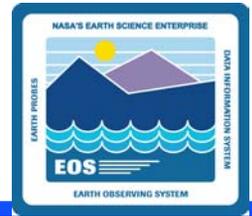
- **Example:**

- » **It would be preferable to run a particular MODIS PGE with the output of a MISR PGE as input**
    - » **However, the MISR output may not be produced every day**
    - » **So the MODIS PGE lists the MISR input as optional with a two-hour timer**
    - » **On those occasions when no MISR output is produced, the MODIS PGE waits for two hours and then is executed without the MISR input**



# Production Requests (Cont.)

- **Illustration of the Alternate Inputs and Optional Inputs Production Rules**
  - PGE has two inputs that are “required” so they must be available for the PGE to be run
  - PGE has one input that is “alternate”
    - » Alternate input can be one of three choices
  - After the pair of required inputs has become available, the alternate inputs are evaluated:
    - » If the primary alternate is available, it is used as input and the PGE is scheduled for execution
    - » One-hour timer on the primary alternate
    - » If the primary alternate is unavailable, the PGE waits until the primary alternate becomes available or the one-hour timer expires, whichever occurs first
    - » If the second alternate is available after the timer for the primary alternate has expired, the second alternate is used as input



# Production Requests (Cont.)

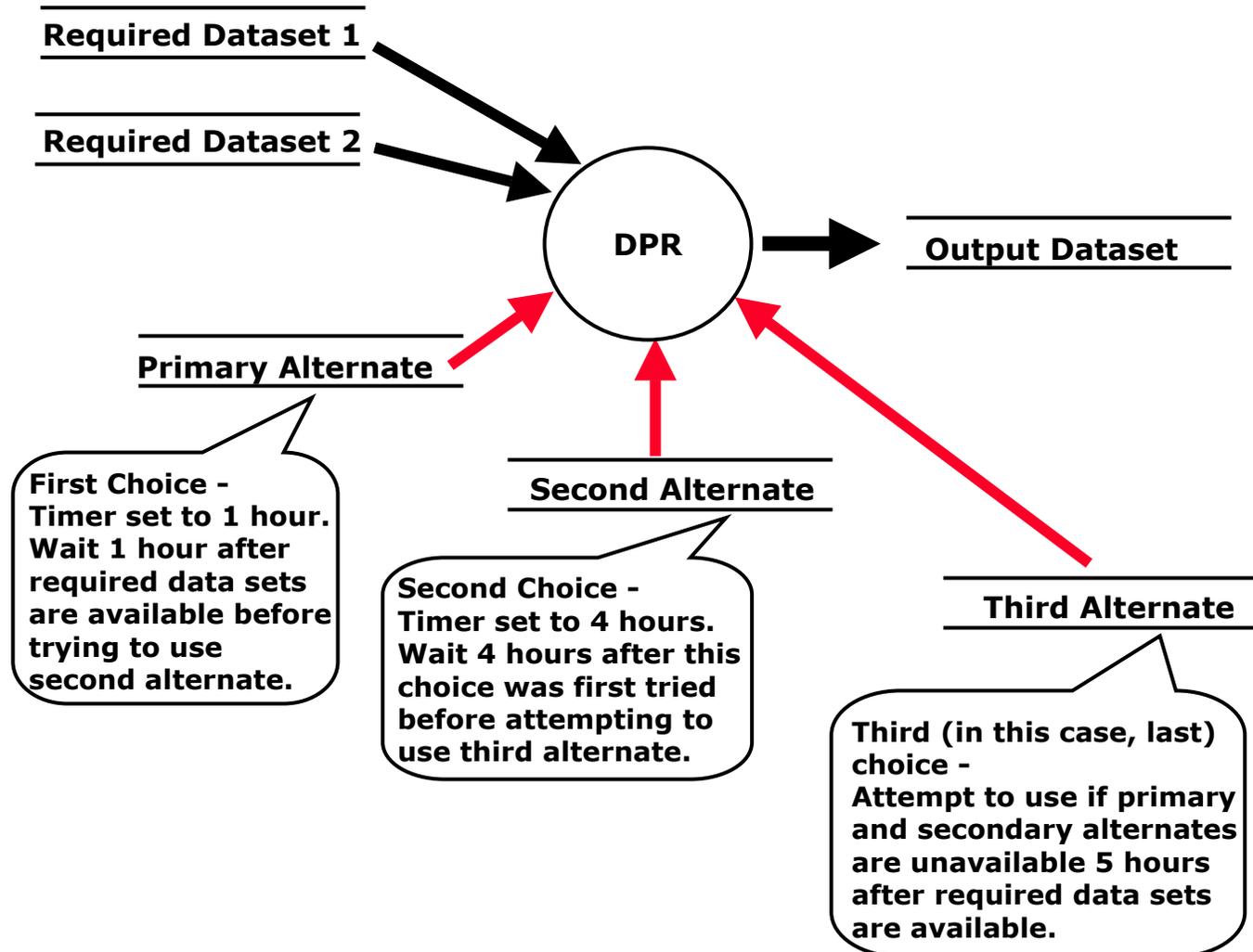
- **Illustration of the Alternate Inputs and Optional Inputs Production Rules (Cont.)**
  - There is a four-hour timer on the second alternate
  - If the second alternate is unavailable, the PGE waits until either the primary alternate or the secondary alternate becomes available or the four-hour timer expires, whichever occurs first
  - If the third alternate is available after the timer for the second alternate has expired, the third alternate is used
  - There is no timer on the third alternate
  - If the third alternate is not available, the PGE waits until either the primary alternate, the secondary alternate, or the third alternate becomes available, whichever occurs first
  - The PGE does not start processing until one of the alternates becomes available



# Production Requests (Cont.)

- **Illustration of the Alternate Inputs and Optional Inputs Production Rules (Cont.)**
  - If instead of an alternate the third input for the PGE had been defined as an optional input...
    - » The preceding scenario would have been the same, except that if neither the primary alternate, the second alternate nor the third option was available after the timers had expired, the PGE would not wait
    - » It would be scheduled for execution without the third input
    - » It would run with the two required inputs only

# Example of Alternate Inputs Production Rule



# Production Requests (Cont.)

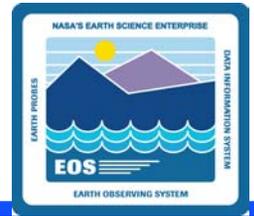


- **Minimum/Maximum Number of Granules Production Rule**
  - **Makes it possible to specify a range of possible granules for a given input or output for a PGE**
  - **Inputs**
    - » **Minimum number of granules the PGE needs for full data coverage**
    - » **Maximum number of granules for the time period**
  - **Outputs**
    - » **Minimum number of outputs that the PGE is expected to produce**
    - » **Maximum number of outputs that the PGE is expected to produce**

# Production Requests (Cont.)



- **Minimum/Maximum Number of Granules Production Rule (Cont.)**
  - **Example:**
    - » **PGE processes data for every 90-minute interval, has a period of 90 minutes, and takes as input a granule with a period of two hours**
    - » **In many instances one granule of the input will satisfy the PGE**
    - » **In other instances, because of the way the two-hour and 90-minute periods overlap, the PGE needs two input granules to cover the time period**
    - » **Minimum Number of Granules = 1**
    - » **Maximum Number of Granules = 2**



# Production Requests (Cont.)

- **Minimum/Maximum Number of Granules Production Rule (Cont.)**
  - Different because it works for both input and output
  - PGE can request of a range of inputs (i.e., 1-10)
  - If a PGE needs at least three granules of an input...
    - » Minimum number of granules is defined as three
    - » PGE is not executed until at least three granules are available
  - Optional outputs are defined when the Minimum Number of Granules is set to zero
    - » PGE can produce none of the particular type of output and still be considered successful
  - If a PGE has a non-zero value for a Minimum Number of Granules associated with an output, and fails to produce any granules of that output type, it is marked as failed

# Production Requests (Cont.)



- **Illustration of Minimum/Maximum Number of Granules Production Rule**
  - PGE processes data related to a one-hour period and takes in both Input 1 and Input 2
  - Input 1 has a PERIOD of 1/2 hour
    - » Every PGE run requires two Input 1 granules
  - Input 2 has a PERIOD of 15 minutes
    - » There are four Input 2 granules for every PGE run
  - PGE produces three Output 1 granules for each run
  - In this case it does not produce any Output 2 granules

# Example of Minimum/Maximum Number of Granules Rule



**Input 1:**

Boundary = Start of Hour

Period = 1/2 hour

**Input 2:**

Boundary = Start of Hour

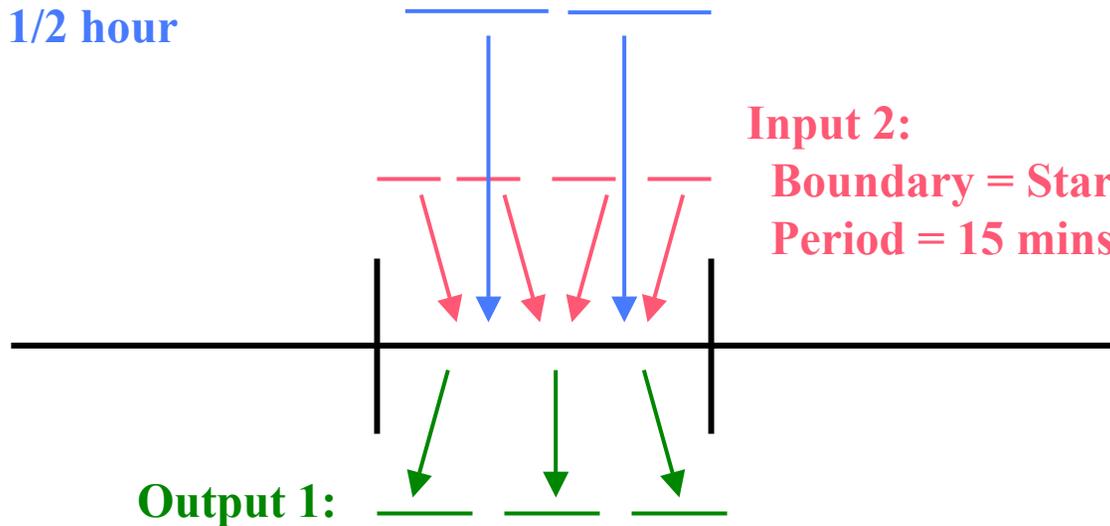
Period = 15 mins

**PGE**

**Output 1:**

**Output 2:**

(No Output)





# Production Requests (Cont.)

- **Illustration of Minimum/Maximum Number of Granules Production Rule (Cont.)**
  - **Minimum and maximum values can affect each input and output**
  - **Input 1:**
    - » **If Minimum Granules is set to anything equal to or less than two for Input 1, the PGE is executed**
    - » **If Minimum Granules is set to three, the PGE is not scheduled (there are not enough Input 1 granules to make the minimum)**
    - » **If Maximum Granules is set to anything equal to or greater than two for Input 1, the PGE is executed**
    - » **If Maximum Granules is set to one, the PGE is not scheduled (there are too many Input 1 granules - the number exceeds the maximum that the PGE can process)**

# Production Requests (Cont.)



- **Illustration of Minimum/Maximum Number of Granules Production Rule (Cont.)**
  - **Input 2:**
    - » **If Minimum Granules is set to anything equal to or less than four for Input 2, the PGE is executed**
    - » **If Minimum Granules is set to five, the PGE is not scheduled (there are not enough Input 2 granules to make the minimum)**
    - » **If Maximum Granules is set to anything equal to or greater than four for Input 2, the PGE is executed**
    - » **If Maximum Granules is set to three, the PGE is not scheduled (there are too many Input 2 granules - the number exceeds the maximum that the PGE can process)**



# Production Requests (Cont.)

- **Illustration of Minimum/Maximum Number of Granules Production Rule (Cont.)**
  - **Output 1:**
    - » If Minimum Granules is set to anything equal to or less than three, the PGE executes successfully
    - » If Minimum Granules is set to four, the PGE is marked as failed (did not produce the expected number of output granules)
    - » If Maximum Granules is set to anything equal to or greater than three, the PGE executes successfully
    - » If Maximum Granules is set to two, the PGE is marked failed (produced too many output granules)
  - **Output 2:**
    - » If Minimum Granules is set to anything other than zero, the PGE is marked failed (did not produce the expected number of output granules)
    - » If Maximum Granules is set to anything equal to or greater than zero for Output 2, PGE is successful



# Production Requests (Cont.)

- **Optional DPRs Production Rule**
  - Also called the **Data-Scheduled Production Rule**
  - Makes the execution of a PGE subject to the availability of a “key input”
  - System generates DPRs for every possible instance of the key input data but executes only the DPRs for which data are either produced in data processing or can be acquired from the archive
  - Applies to PGEs that process certain kinds of non-routine data
    - » **Routine Data:** Data that can be predicted, that come in at specific intervals and are always of a specified length
    - » **Non-Routine Data:** Data that cannot be predicted because they come in at random periods and/or their length is variable (Examples include an "optional" output of an upstream PGE, or data that are archived at random periods; e.g., some forms of ASTER data)

# Production Requests (Cont.)



- **Optional DPRs Production Rule (Cont.)**
  - An Optional DPR has as its key input a non-routine data type
  - There are two sets of circumstances that lead to the scheduling of Optional DPRs:
    - » Every possible time that the input is produced in data processing (i.e., the key input is produced as an "optional" output by an upstream PGE)
    - » Whenever a new granule (of a particular data type) can be acquired from the archive (e.g., archived data that were inserted at unpredictable times)



# Production Requests (Cont.)

- **Optional DPRs Production Rule (Cont.)**
  - **Example 1: Key input is produced as an "optional" output by an upstream PGE**
    - » **One MODIS PGE produces a certain product only when the input data were collected during the satellite's "Day" mode**
    - » **A second MODIS PGE is scheduled to use the optional ("Day"-mode) product from the first MODIS PGE as its key input**
    - » **Second MODIS PGE is scheduled to run after every instance of the first MODIS PGE**
    - » **Only the DPRs that can use the optional products resulting from runs of the first MODIS PGE are executed**
    - » **The remaining DPRs cannot be executed because there is no input data for them**



# Production Requests (Cont.)

- **Optional DPRs Production Rule (Cont.)**
  - **Example 2: Key input granules were inserted into the archive at unpredictable times**
    - » **ASTER routine processing**
    - » **DAAC ingests and archives ASTER production data from tapes supplied by the ASTER Ground Data System on a frequent but not predictable basis**
    - » **When creating a Production Request, the Production Planner specifies the insertion time range as opposed to the collection time**
    - » **Insertion time range = time period when the desired data were archived**
    - » **Collection time = when the satellite instrument gathered the data**
    - » **DPRs specifying the ASTER PGE are scheduled and executed for the data granules that were actually inserted in the archive during the specified time range**



# Production Requests (Cont.)

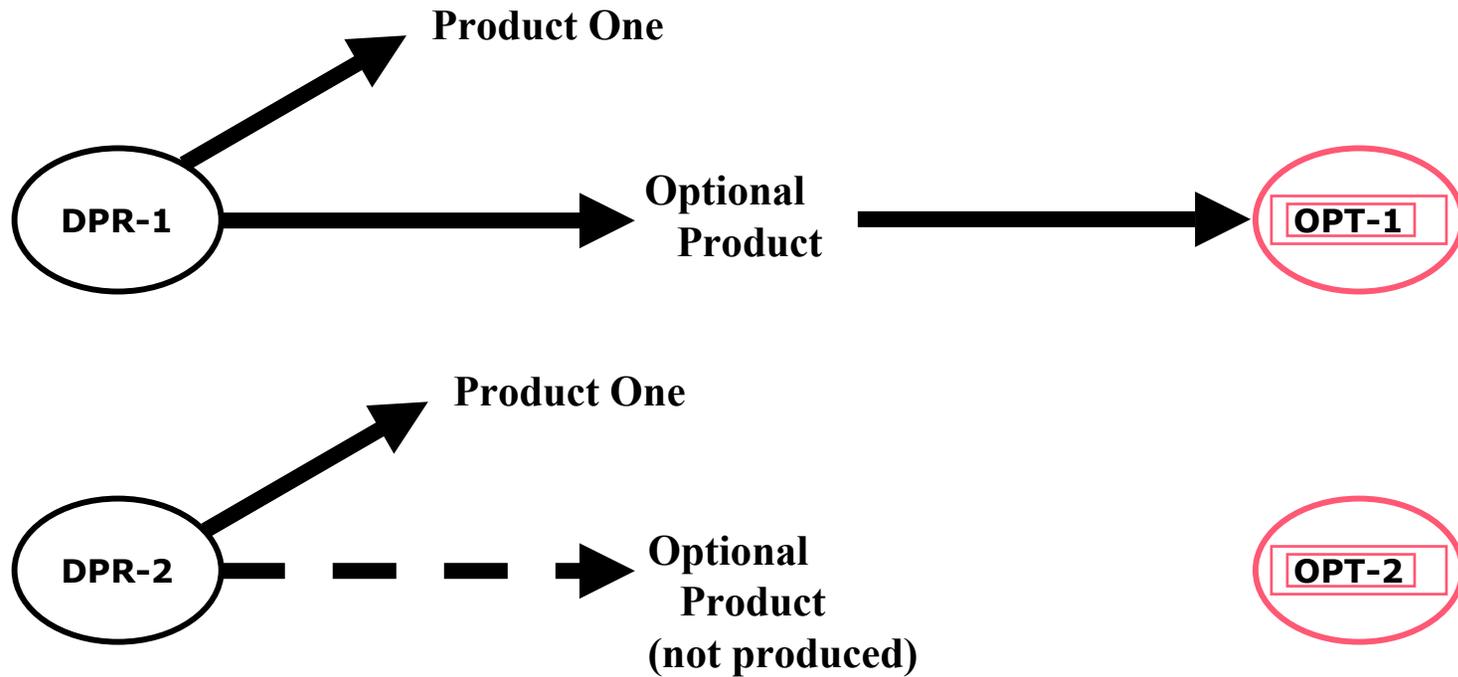
- **Illustration of the Optional DPRs Production Rule**
  - Two DPRs (i.e., DPR-1 and DPR-2) for the upstream PGE
  - Two DPRs (i.e., OPT-1 and OPT-2) for the PGE subject to the Optional DPRs Production Rule
  - “Optional DPRs” PGE takes as input the optional output of the upstream PGE
  - When it is executed, DPR-1 produces the optional output, so the dependent DPR (OPT-1) is executed
  - DPR-2 (on which OPT-2 depends) does not produce the optional output so OPT-2 is not executed

# Example of Optional DPRs Production Rule



## Upstream PGE

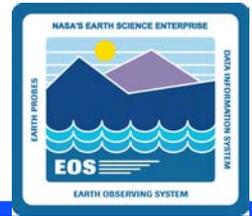
## Optional DPRs



# Production Requests (Cont.)



- **Metadata Checks and Metadata Query Production Rules**
  - Similar in definition and use
  - Both allow the PGE to specify granule-level metadata values that define whether the PGE can accept one (or more) of its inputs
  - Difference is in the results of metadata search



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Metadata Checks Production Rule**
    - » When PLS requests the Science Data Server to search for the input(s), the Science Data Server "checks" the metadata of all granules that match the time frame with respect to the allowed value(s)
    - » If any granule fails to match the specified value(s), the PGE is not executed
  - **Metadata Query Production Rule**
    - » When PLS requests the Science Data Server to search for the input(s), the Science Data Server adds to the query the PGE's desired metadata value(s)
    - » Only the granules that match the time frame of the PGE plus the granule-level metadata value(s) specified by the PGE are staged as input to the PGE
    - » If no granules are found matching the conditions and the input is not optional, the PGE is not executed



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Example of Metadata Checks:**
    - » **A MODIS PGE is run when the Percent Cloud Cover of its inputs is greater than 25 percent**
    - » **The Metadata Checks Production Rule is used to specify the granule-level metadata value of greater than 25**
    - » **When the PGE is scheduled and is ready to start, two granules match the timeframe of the Production Request for the input with the Metadata Check**
    - » **If both granules have a Percent Cloud Cover greater than 25 percent, execution of the PGE starts and both granules are staged**
    - » **If one of the granules has a Percent Cloud Cover of 15 percent, the PGE is not executed**



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Example of Metadata Query:**
    - » **A MODIS PGE is run when as many granules as possible of one of its inputs have a QA Value = "Good"**
    - » **The Metadata Query Production Rule is used to specify the granule-level metadata value = "Good"**
    - » **When the PGE is scheduled and is ready to start, two granules match the time frame of the production request for the input with the Metadata Query**
    - » **If both granules have a QA Value = "Good", execution of the PGE starts and both granules are staged**
    - » **If one of the granules has a QA Value = "Bad", the PGE executes but with only one granule (the one with QA Value = "Good")**



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Metadata Checks and Metadata Query Production Rules are used in conjunction with the times specified in the Basic Temporal Production Rule or other production rules**
  - **Metadata Check or Query is added information that further refines what granules are sought by the PGE**



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **For past data**
    - » **Production Request Editor performs the Metadata Query or Metadata Check immediately**
    - » **Depending on what other production rules may be applicable, if no data is found to match the “query” (or data is found that does not match the “check”), the DPR fails to be created**
  - **For future data**
    - » **Metadata Query/Metadata Check is put off until the DPR Data Collection Stop Time passes plus the value defined in the ODL for QUERY\_DELAY**
    - » **Delay allows the Metadata Query/Metadata Check to be put off until it's likely that all matching data will be present**



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Multi-Granule ESDTs are a special case of the Metadata Query Production Rule**
    - » **Used for PGE inputs or outputs when more than one granule of the same ESDT exists for the same temporal range (time period)**
    - » **Multi-Granule ESDT mechanism employs a metadata parameter to differentiate between "equal in time" granules**



# Production Requests (Cont.)

- **Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **Data Day is an addition to the Metadata Query Production Rule**
    - » **Involves runtime parameter values**
    - » **Two settings (i.e., “Start Data Day” and “End Data Day”) allow a PGE to perform a Metadata Query for the start of the Data Day and the end of the Data Day**
    - » **The Start Data Day and End Data Day are calculated by subtracting twelve (12) hours from the starting day for which the PGE is executing and adding twelve (12) hours onto the ending day for which the PGE is running. So if the PGE is running from 00:00:00 on 07/04 to 00:00:00 07/05 then the Start Data Day = 07/03 12:00:00 and the End Data Day = 07/06 12:00:00.**



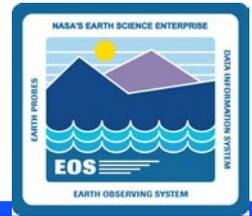
# Production Requests (Cont.)

- **Illustration of the Metadata Checks and Metadata Query Production Rules**
  - If no Metadata Check or Query were applicable, a particular PGE would use three granules of input (i.e., Granules A through C)
  - However, in the illustration the metadata value `%CloudCover` is to be checked/queried
    - » Granules A through C each have a different value for `%CloudCover`
  - There can be more than one Metadata Check or Metadata Query on a given input
    - » A Metadata Check on `%CloudCover` can be combined with a Metadata Query on another parameter to further limit the input



# Production Requests (Cont.)

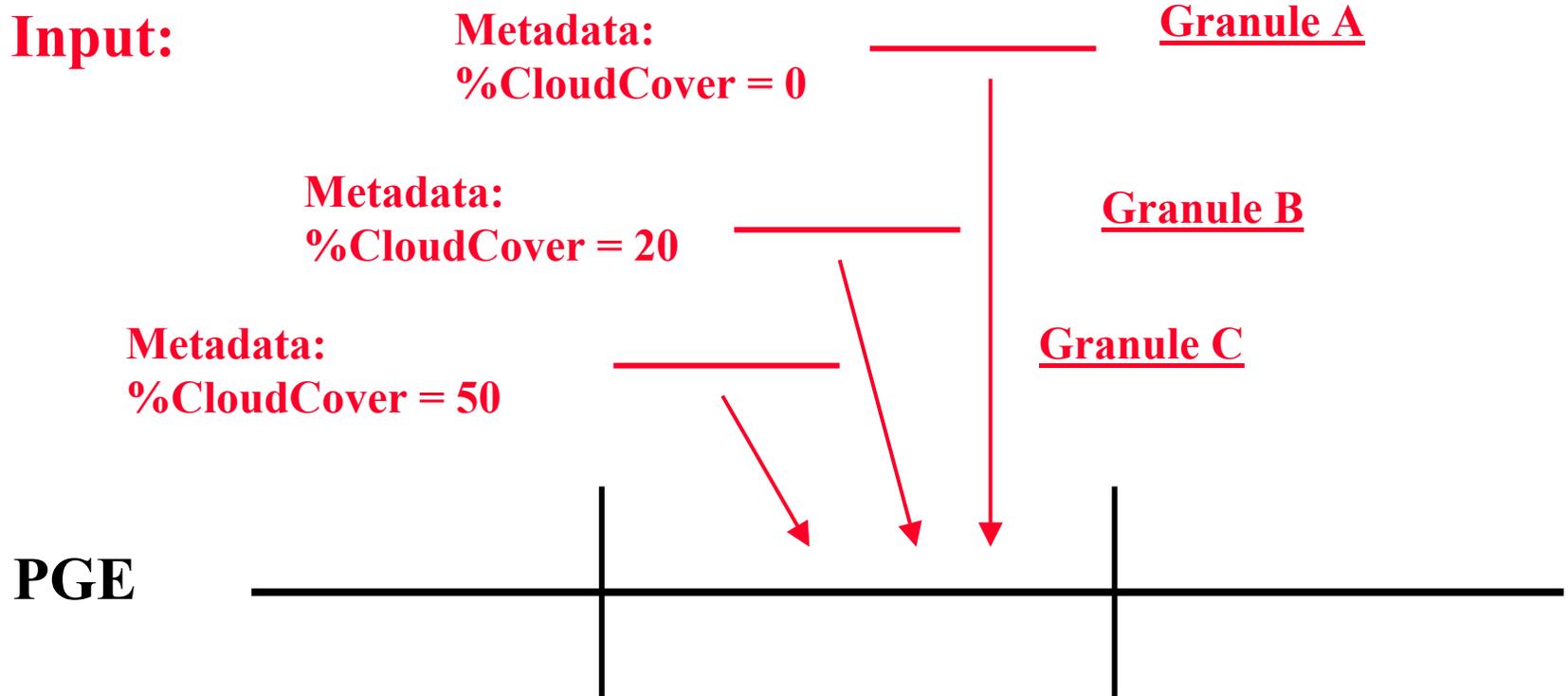
- **Illustration of the Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **The following results demonstrate differences between Metadata Checks and Metadata Query:**
    - » **Metadata Check of %CloudCover < 80: All three granules are acquired and the PGE is executed**
    - » **Metadata Query of %CloudCover < 80: All three granules are acquired and the PGE is executed**
    - » **Metadata Check of %CloudCover = 50: PGE is not scheduled because only one of the three granules (Granule C) meets the criterion**
    - » **Metadata Query of %CloudCover = 50: Granule C is found and if the PGE's Min/Max Granules parameters are set to allow one granule, that one granule is acquired and the PGE is executed**



# Production Requests (Cont.)

- **Illustration of the Metadata Checks and Metadata Query Production Rules (Cont.)**
  - **The following results demonstrate differences between Metadata Checks and Metadata Query:**
    - » **Metadata Check of %CloudCover = 20: PGE is not scheduled because only one of the three granules (Granule B) meets the criterion**
    - » **Metadata Query of %CloudCover = 20: Granule B is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is executed**
    - » **Metadata Check of %CloudCover < 20: PGE is not scheduled because only one of the three granules (Granule A) meets the criterion**
    - » **Metadata Query of %CloudCover < 20: Granule C is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is executed**

# Example of Metadata Checks/Query



# Production Requests (Cont.)



- **Spatial Query Production Rule**
  - PGE selects input(s) based on the spatial coverage of another input (called the key input)
  - PDPS queries the Science Data Server for the spatial coverage of the key input
  - PDPS uses key input's spatial coverage in acquiring any subsequent inputs that the PGE has requested that have the same spatial coverage
  - Without specifying coordinates, PDPS can match inputs against the spatial constraint of the key input, and give to a PGE only those granules which overlap in area

# Production Requests (Cont.)

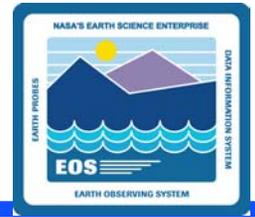


- **Spatial Query Production Rule (Cont.)**

- **Example:**

- » **Level 0 input data for an ASTER DPR covers a small section of the Earth**
    - » **The PGE requires ancillary data that covers the same area to complete its processing**
    - » **The PGE uses the Spatial Query Production Rule to mark the geographic input as its key input**
    - » **The PGE specifies that the ancillary input is to be retrieved for the same spatial coverage as that of the key input**
    - » **When PDPS finds an input granule for the PGE, it performs a Spatial Query to acquire the ancillary input with the same spatial coverage as that of the key input**

# Production Requests (Cont.)



- **Spatial Query Production Rule (Cont.)**
  - **Spatial Pad is an addition to the Spatial Query Production Rule**
    - » **Spatial Pad is a means of padding the spatial constraints of the key input**
    - » **The specified pad is added to all sides of the key input's spatial shape**
    - » **All granules that intersect the expanded area are retrieved**



# Production Requests (Cont.)

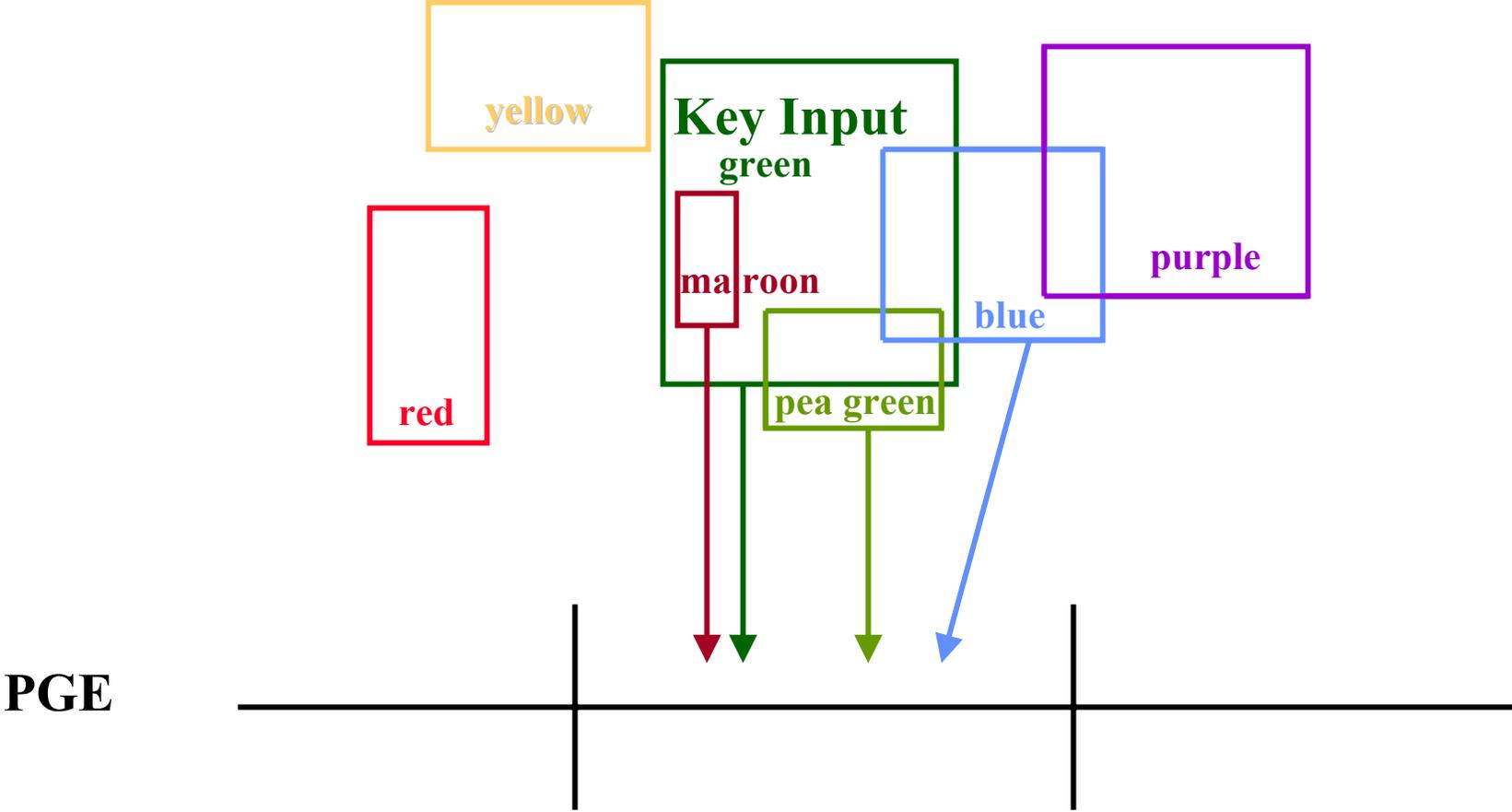
- **Illustration of the Spatial Query Production Rule**
  - **PGE has two input types**
    - » **One is the key input**
    - » **The other type of input has granules labeled with the names of various colors**
  - **One granule (i.e., “green”) of the key input is found**
  - **Spatial coordinates of the granule are retrieved**
  - **All inputs of the second ESDT are checked for overlap with the key input’s coordinates**



# Production Requests (Cont.)

- **Illustration of the Spatial Query Production Rule (Cont.)**
  - **Assuming that all granules relate to the same time period, the granules are evaluated as follows:**
    - » **“yellow” granule is not retrieved as an input (spatial coordinates do not overlap with those of the key)**
    - » **“red” granule is not retrieved as an input (same as “yellow”)**
    - » **“blue” granule is retrieved as an input (spatial coordinates overlap with those of the key input)**
    - » **“maroon” granule is retrieved as an input (same as “blue”)**
    - » **“pea green” granule is retrieved as an input (same as “blue”)**
    - » **“purple” granule is not retrieved as an input (same as “yellow”)**

# Example of Spatial Query

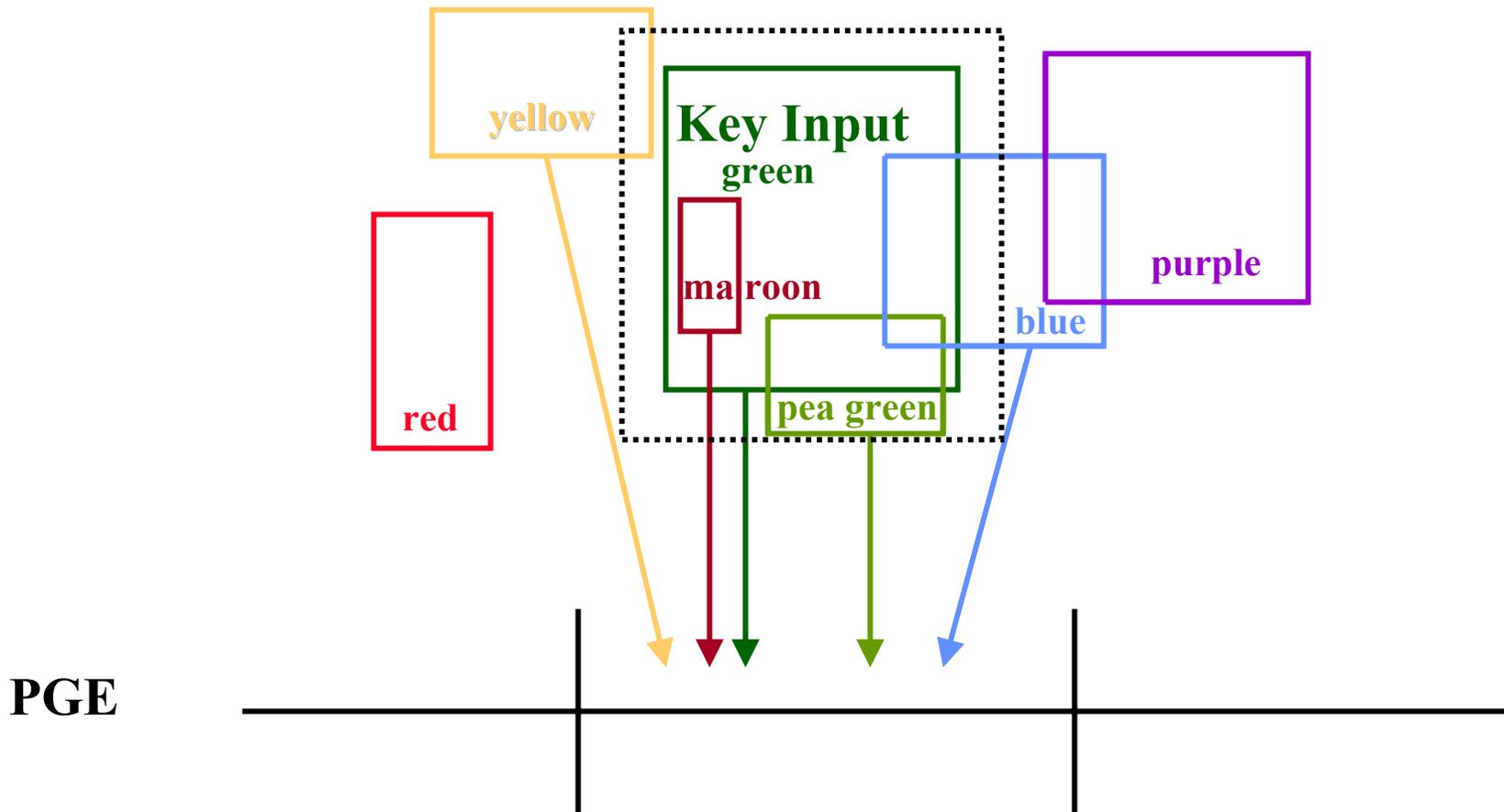
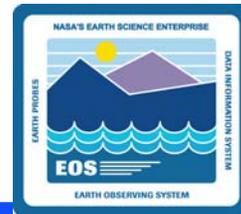




# Production Requests (Cont.)

- **Illustration of Spatial Query with Spatial Pad**
  - Assuming that all granules relate to the same time period, the granules are evaluated as follows:
  - period, the granules are evaluated as follows:
    - » “yellow” granule is retrieved as an input (spatial coordinates do not overlap with those of the key but do overlap the padded area)
    - » “red” granule is not retrieved as an input (spatial coordinates do not overlap with either the key or pad)
    - » “blue” granule is retrieved as an input (spatial coordinates overlap with the key)
    - » “maroon” granule is retrieved as an input (same as “blue”)
    - » “pea green” granule is retrieved as an input (same as “blue”)
    - » “purple” granule is not retrieved as an input (same as “red”)

# Example of Spatial Query with Spatial Pad



# Production Requests (Cont.)



- **Closest Granule Production Rule**
  - PGE requests the nearest input granule from the Data Processing Request time
  - PDPS requests a search forward or backward for a specified period of time until it finds a granule that matches the request; however, ...
    - » Number of queries performed is limited
    - » Period length of the query is limited
  - **Supersedes the Most Recent Granule Production Rule**
    - » Allowed the search for inputs to go back only, not forward

# Production Requests (Cont.)



- **Closest Granule Production Rule (Cont.)**
  - **Example:**
    - » **A PGE processes data at daily intervals and could use a particular type of calibration granule that would allow it to determine the nearest parameters of the instrument**
    - » **Although most calibration coefficients are defined as static granules, in this case there is a dynamic granule that is received about once a month**
    - » **The closest such granule would be optimal, so the PGE uses the Closest Granule Production Rule to search forward or backward from the time of the DPR to find the nearest calibration granule**

# Production Requests (Cont.)



- **Closest Granule Production Rule (Cont.)**
  - **Three values determine the period of the query**
    - » **Period (Offset)**
    - » **Direction**
    - » **Maximum Number of Queries**

# Production Requests (Cont.)



- **Closest Granule Production Rule (Cont.)**
  - **Period (Offset)**
    - » Tells the PDPS software the query duration
  - **Direction**
    - » Indicates whether the query goes forward (positive) or backward (negative) in time
    - » In the `PIDataTypeReq` table in the PDPS database, the **Direction** and the **Period** information are combined, so the sign of value (+ or -) indicates the **Direction** and the magnitude indicates the **Period**
  - **Maximum Number of Queries**
    - » Tells the PDPS software how many time periods (as defined by the **Offset**) to search (either forward or backward in time) for a matching granule



# Production Requests (Cont.)

- **Closest Granule Production Rule (Cont.)**
  - PDPS does a Basic Temporal query before using Closest Granule to find the input
  - If the desired input is not found within the time period of the DPR, PDPS performs a query (in the specified direction) against the Science Data Server for the period defined by the offset
  - If no matching granule is found, PDPS repeats the query, going backward or forward in time by the value specified in the offset
  - If no acceptable granule has been found before the maximum number of queries is reached, PDPS fails to generate the DPR due to insufficient input data
  - special case for the forward search:
    - » when the next search interval exceeds the current time, the search stops at the current time

# Production Requests (Cont.)



- **Closest Granule Production Rule (Cont.)**
  - If the DPR is planned for a future time, the DPR is created using place-holder granules and a timer is activated
  - When the current time reaches the stop time of DPR, the timer invokes the Closest Granule method to search for the actual granules, which are then used to replace the dummy granules

# Production Requests (Cont.)



- **Illustration of the Closest Granule Production Rule**
  - PGE boundary is “start of day”
  - PGE period is one hour
    - » PGE is scheduled to run for one hour’s worth of input data
  - Input period is one hour
    - » Can come in at any hour of the day
  - PGE requests one granule of input
  - Offset is 6
    - » Queries in six-hour intervals
  - Direction is backward
  - Maximum number of queries is two

# Production Requests (Cont.)



- **Illustration of the Closest Granule Production Rule (Cont.)**
  - PDPS performs a query for the input based on the time period of the DPR
    - » No matching data is found
  - PDPS uses the Closest Granule information to query for a six-hour period beginning six hours before the start time of the DPR
    - » Again nothing is found
  - A second Closest Granule query is performed, this one six hours before the last Closest Granule query
    - » Two granules are found that match the query
  - PDPS selects the granule that is later in time and schedules the PGE to use it as input

# Production Requests (Cont.)



- **Illustration of the Closest Granule Production Rule (Cont.)**
  - **If the Closest Granule Production Rule were used in conjunction with the Minimum/Maximum Number of Granules Production Rule, it might be possible for both granules to be selected in the previously described Closest Granule query**
    - » **If the example included setting the Maximum Number of Granules to two, both granules would be selected as input to the PGE**

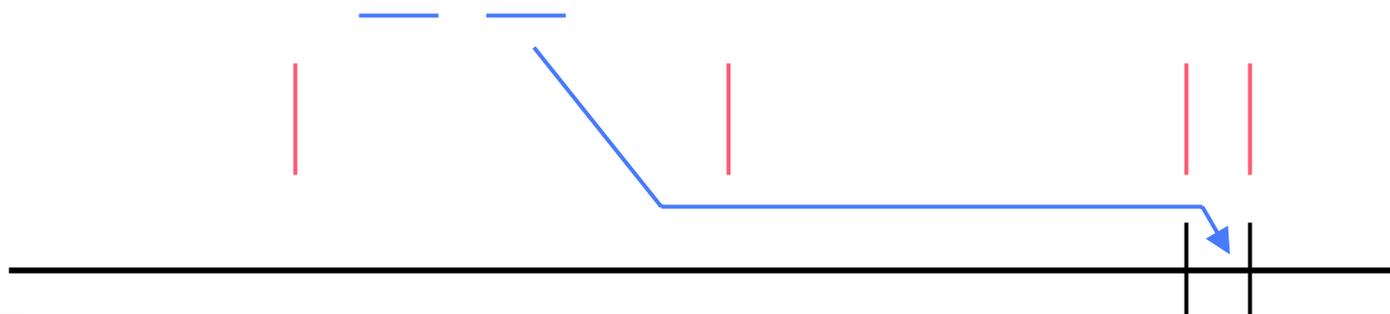
# Example of Closest Granule



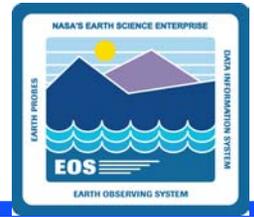
Input: Boundary = Start of Hour  
Period = 1 hour

PGE:

Boundary =  
Start of Day  
Period =  
1 hour



Closest Granule Query:  
Query Period = -6 hours  
Max Queries = 2



# Production Requests (Cont.)

- **Orbital Processing Production Rule**
  - **Uses the orbit of the spacecraft to determine the time period for the inputs and outputs of the PGE**
  - **Example:**
    - » **A PGE processes Level 0 data related to each orbit of the Terra satellite**
    - » **The Terra satellite has an orbital period of 98 minutes so the PGE is scheduled to process data for each 98-minute interval**
    - » **Since Level 0 data are received every two hours, the data staged for the PGE include every Level 0 granule that falls within the 98-minute PGE interval**
    - » **Only one granule of Level 0 data is relevant to some 98-minute orbits**
    - » **Two granules of Level 0 data are relevant to other 98-minute orbits**



# Production Requests (Cont.)

- **Orbital Processing Production Rule (Cont.)**
  - Uses the “period” and “boundary” concepts just like the Basic Temporal Production Rule
    - » Orbit of the spacecraft is taken into account when a PGE or its data are marked as orbit scheduled
  - When responding to a Production Request for orbit-scheduled processing, PDPS determines the orbit of the satellite via information from SSI&T
  - Information in the PDPS database gives the start time and length of a particular orbit or set of orbits
  - PDPS extrapolates (or interpolates in the case of an orbit between two orbital periods stored in the database) the start and end times of the PGE that is specified in the Production Request
  - Data are sought on the basis of the derived start and stop times and the appropriate data granule(s) is/are staged before the PGE is executed



# Production Requests (Cont.)

- **Orbital Processing Production Rule (Cont.)**
  - **Orbit model is a model of the satellite's orbits that allows PDPS to perform extrapolations for the Orbital Processing Production Rule**
    - » **Model is a combination of a database table and a simple algorithm**
    - » **Database table stores Orbit Number/Orbit Start Time/Orbital Period combinations**
    - » **Algorithm uses the data to compute the same type of data relevant to subsequent orbits**
    - » **Works by extrapolation (unable to calculate data for any orbit that precedes the earliest entry in the database table)**
    - » **Data for the Orbit Model is specified in the Orbit Model ODL file, which is read only if the PGE requires orbital information**



# Production Requests (Cont.)

- **Orbital Processing Production Rule (Cont.)**
  - **Orbital path is the path of the satellite over the Earth**
    - » **Number from 0-233 that indicates the region of the Earth covered by a particular orbit**
    - » **Because of the implementation of Orbital Path, there needs to be a mapping between the orbital path calculated by PDPS and the orbital path number expected by the PGEs**
  - **Runtime parameters can be set to values associated with Orbital Processing**
  - **The following orbital parameters can be placed under runtime parameters:**
    - » **Orbit Number**
    - » **Orbital Path Number**
    - » **Orbit Number within the Day**
    - » **Granule Number within the Orbit**

# Production Requests (Cont.)



- **Orbital Processing Production Rule (Cont.)**
  - **Orbit Number**
    - » Number of the orbit (starting from zero) and continually increasing
  - **Orbital Path Number**
    - » Number of the path that maps to the orbit number
    - » Orbital path number is the 0-233 orbital path traversed by the satellite
  - **Orbit Number within the Day**
    - » Number of the orbit within the given day
    - » Includes any orbit that starts within the given day
  - **Granule Number within the Orbit**
    - » Number of the granule within a given orbit
    - » Includes any granule that starts within the given orbit



# Production Requests (Cont.)

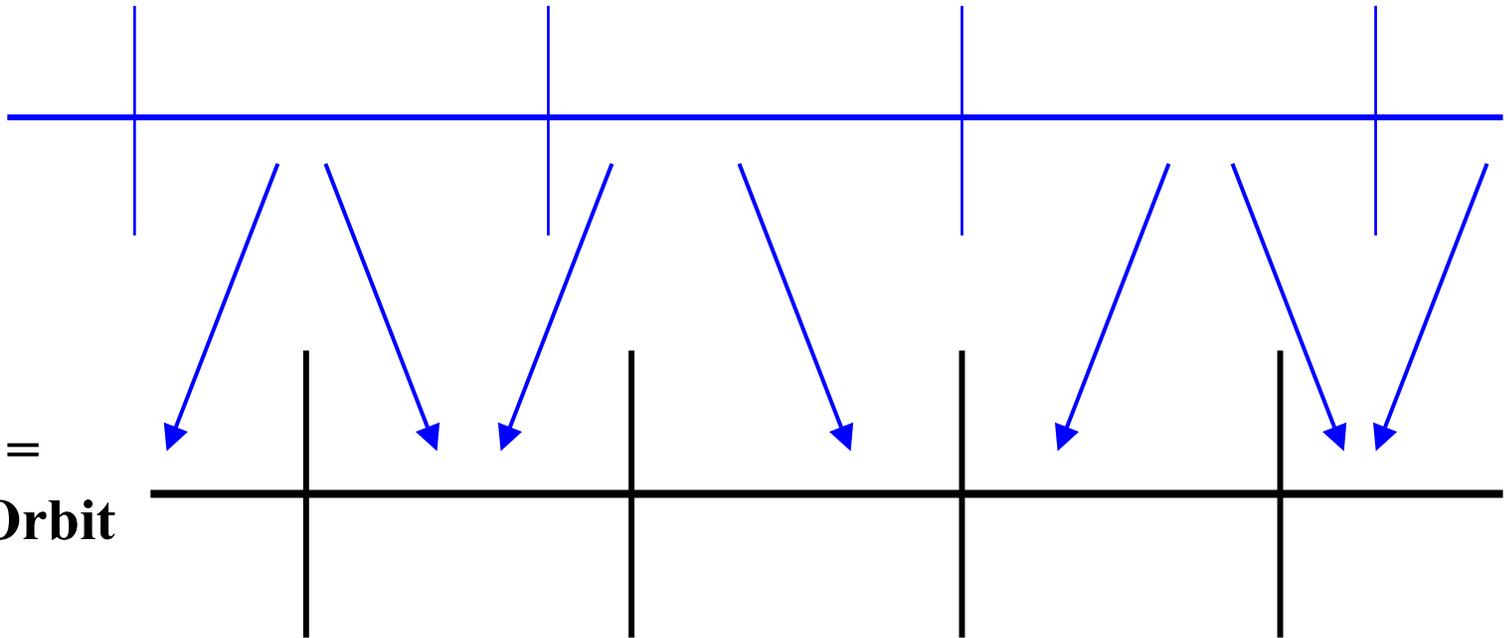
- **Illustration of the Orbital Processing Production Rule**
  - **PGE takes a two-hour input, but is scheduled based on the orbit time and period of the satellite**
  - **PDPS uses the data collected at SSI&T to predict the time of the orbit and performs the query to the Science Data Server for the input based on that extrapolated or interpolated orbital time**
  - **Granules of input data are allocated to DPRs based on their ability to cover the DPR's time period**
    - » **Length of an orbit is less than the period of the two-hour input**
    - » **Sometimes a single granule may cover the input time range of a PGE execution**
    - » **At other times two granules are required**
    - » **Rule would work equally well if the data were of a shorter period than the orbit of a satellite**

# Example of the Orbital Processing Production Rule



**Input: Boundary = Start of Hour**  
**Period = 2 hours**

**PGE:**  
**Boundary =**  
**Start of Orbit**  
**Period =**  
**1 orbit**





# Production Requests (Cont.)

- **Multiple DPRs for Insertion Time Production Rule**
  - **Allows the creation of DPRs for multiple granules with the same insertion time**
    - » **Affects ASTER L1B routine processing only**
  - **Is implemented when the Production Planner creates a production request**
    - » **On the Production Request Editor the Production Planner enters the Duration information (Begin and End date and time) for the Insertion Time (versus Collection Time) and then ensures that the Multiple DPRs toggle button is depressed**
    - » **If the Multiple DPRs toggle button is not depressed, the default rule is applied and when the production request is saved a DPR is created for the most recent granule with the specified insertion time**



# Production Requests (Cont.)

- **Multiple DPRs for Insertion Time Production Rule**
  - **Applies to the key input (e.g., ASTER L1B) only**
    - » It does not affect any ancillary data types (for which the most recent granule is selected)
  - **When the Multiple DPRs for Insertion Time production rule is invoked, a DPR is generated for each key-data-type granule returned from the insertion-time query**
    - » Therefore, if multiple key-data-type granules are returned for the same collection period, multiple DPRs are generated
    - » The DPR IDs are different in the last three characters, which are generated randomly



# Production Requests (Cont.)

- **Tiling Production Rule**
  - **PGE is run for a series of specific geographic locations called "tiles"**
  - **Tiles are defined before the PGE is scheduled, specifying the longitude and latitude of four points that outline each tile**
  - **When the PGE is scheduled, it is scheduled for an entire day**
    - » **Data is queried based on both a time frame and the geographic location specified**
  - **Each run of the PGE for the day is for a specific tile, and only data that overlap or fit within the geographical coordinates of the tile are staged for the PGE**

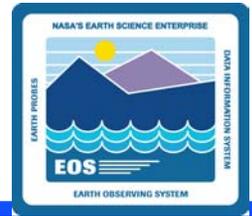


# Production Requests (Cont.)

- **Tiling Production Rule (Cont.)**

- **Example:**

- » **A MODIS PGE is designed to run on data for a specific geographic location every day**
    - » **The location is expressed as a polygon defined by latitude and longitude coordinates**
    - » **The MODIS PGE is scheduled every day, and data are retrieved that match the time period (the day for which the PGE is being executed) and some part of it falls within the geographic constraints of the tile**
    - » **The PGE runs and produces data that define information about the particular tile**
  - **Period and boundary are used to specify the timing of input data and provide indications of how often the PGE should be executed**
  - **At least some of the input data are retrieved on the basis of the coordinates defined for the tile on which the PGE is executing**



# Production Requests (Cont.)

- **Tiling Production Rule (Cont.)**
  - **There are really two kinds of tiling:**
    - » **PGE takes in data based on geographic shapes (tiles) and produces an output or outputs for the specified geographical coverage**
    - » **PGE takes in an already tiled product as input (This form of tiling is more like a Metadata Query using a runtime parameter value to acquire the correct tiled data)**
  - **Runtime parameters can be set to the ID of the tile being processed**
    - » **Since PDPS schedules a Tiling PGE to run once per tile, it can pass the identifier of the tile to the PGE**
    - » **The identifier can be placed under a specified runtime parameter in the PCF, or it can be used in a Metadata Query for a PGE that would use already tiled data as input**



# Production Requests (Cont.)

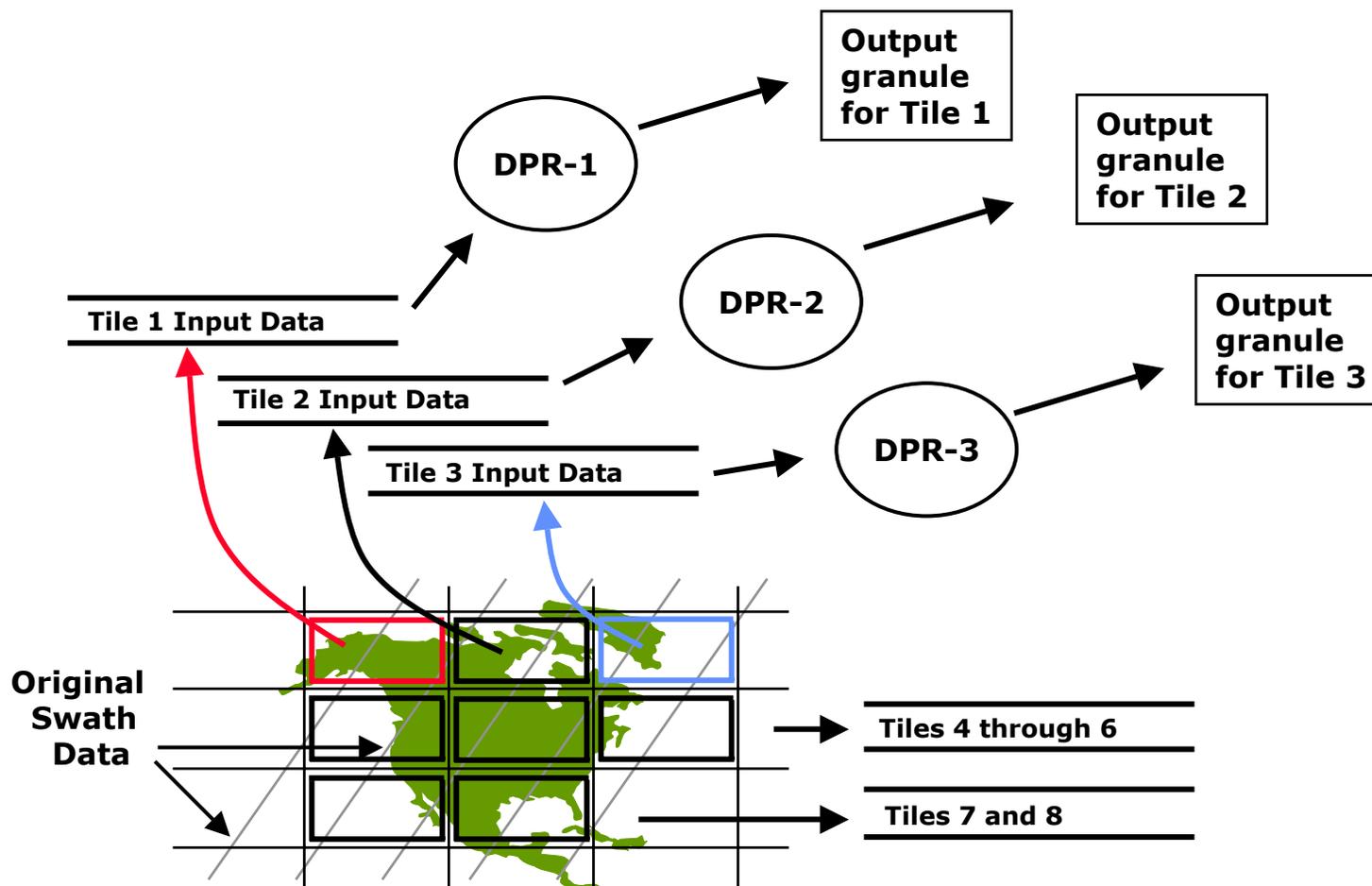
- **Illustration of the Tiling Production Rule**
  - PGE runs once per defined tile
  - For every tile in the Tile Scheme a Data Processing Request is created to run using data that match the geographic extent of the tile
  - PDPS sends the coordinates of the tiles (e.g., Tiles 1 through 3) to the Science Data Server when requesting data
  - Only the granules that fall fully or partially within the defined tile are acquired
  - The PGE itself must be set up to handle the fact that the entire area of the tile may not be covered by available data



# Production Requests (Cont.)

- **Illustration of the Tiling Production Rule (Cont.)**
  - **PGE must set the metadata of the output products so a downstream Tiling PGE can acquire the correct granules for a given tile**
    - » **PDPS does not keep track of tiles once they have been produced**
  - **PDPS matches up the granules needed for a downstream PGE via a query to the Data Server Subsystem**

# Example of Tiling



# Production Requests (Cont.)



- **Possible future enhancements to the Tiling Production Rule:**
  - **Zonal Tiling**
    - » Supports tiles that cover a band around the Earth between two given latitudes
  - **Tile Clustering**
    - » Involves grouping tiles that cover nearby geographic locations together so that data that span the tiles may be staged only once
    - » Intended to improve the performance of Tiling
    - » Also provides for the ability to prioritize one group of tiles over others (so specific geographic outputs are produced before other geographic outputs)

# Production Requests (Cont.)



- **Production Planning Considerations**

- **Production Request (PR) is a template request to generate a particular data product**
  - » **results in a production run of the associated SCF-provided PGE**
- **PR specifies a range over which the data products are to be produced or the PGEs are to be scheduled**
  - » **time**
  - » **snapshot**
  - » **data**
  - » **orbit**



# Production Requests (Cont.)

- **Production Planning Considerations (Cont.)**

- **During normal operations Production Planner should not have to add PRs to the PDPS database very frequently**
  - » **PR might request that the data product be produced for only a single day's data**
  - » **PR might request that data products be produced for every opportunity of input data for several months**
  - » **Early in a mission the SCF may prefer to request processing for a short time period only**
  - » **SCF reviews the quality of the products and notifies the DAAC of the need for any changes to the PR**
  - » **When the SCF has developed a good understanding of the instrument's behavior, the team may be comfortable requesting processing for months at a time**



# Production Requests (Cont.)

- **Production Planning Considerations (Cont.)**

- **DAAC operations may have operational reasons for wanting to issue processing requests for a limited time period**
- **Production Planner has to balance the various considerations when determining whether or not to create or update a PR**
- **Planning decisions are made on the basis of locally defined planning strategies for supporting the SCFs' data processing needs**
- **Production Planner must coordinate with the Resource Planner**
  - » **before planning production**
  - » **resolve all resource allocation issues**
  - » **determine what resources are available for use in processing**



# Production Requests (Cont.)

- **Chain Heads**

- **A chain is a related set of DPRs in which the output of the first DPR in the chain (i.e., the “chain head”) is used as input for at least one subsequent DPR**
  - » **Output of the subsequent DPR(s) may be used as input for further DPR(s) and so on until the other end(s) of the chain is (are) reached**
  - » **Chained DPRs must be processed in the proper sequence from the chain head to the chain’s other end(s)**
- **By default all DPRs in a chain are executed on a single computer**
  - » **Most of the data for the chain are produced and consumed locally, so communication of data on network-mounted file systems is kept to a minimum**



# Production Requests (Cont.)

- **Chain Heads (Cont.)**

- **Production Request Editor provides a means by which the Production Planner can identify a PR as the first in a chain of PRs**
- **The outputs of the DPR(s) that PLS creates from the chain-head PR are used as inputs to one or more subsequent DPR(s) specified in other PRs**
  - » **Any downstream DPR (i.e., that requires the output of a preceding DPR as its input) is known as a "child" DPR**
  - » **Any DPR that provides output for use as an input to a child DPR is a "parent" DPR (chain heads are parent DPRs)**
  - » **To completely deactivate chain processing, declare every PGE to be a chain head**
  - » **To enable total chain processing, declare no PGE to be a chain head**



# Production Requests (Cont.)

- **Chain Heads (Cont.)**

- The "Determine Chain" script (i.e., `EcPIDetermineChain.pl`) finds chains starting with the DPREP PGEs
- In addition to designating chain heads in a PR, the Production Planner may select a particular virtual computer for running the chain-head DPRs
  - » Selection of a virtual computer is optional
  - » If no machine is specified, the system tries to schedule the DPR on the machine where the bulk of its accepted inputs (both static and dynamic inputs) are staged



# Production Requests (Cont.)

- **Chain Heads (Cont.)**
  - **Implementation of chain heads includes ....**
    - » **Addition of chainFlag and scheduledMachine columns to the PIDataProcessingRequest and PIProductionRequest database tables in the PDPS database**
    - » **Addition of chainId column to the PIDataProcessingRequest database table in the PDPS database**
    - » **The Production Request Editor sets the chainFlag and scheduledMachine values based on Production Planner input when a PR is created**
    - » **It is also possible to set the chainFlag value using the Planning Workbench GUI at plan activation time for DPRs that have no "parent" DPRs producing data used for their input**



# Production Requests (Cont.)

- **Chain Heads (Cont.)**

- **The Planning Workbench GUI calls the "Determine Chain" script when a plan is activated**

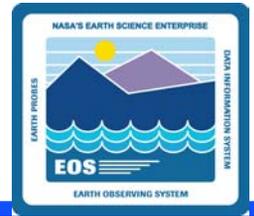
- » ***/usr/ecs/MODE/CUSTOM/bin/PLS/EcPIDetermineChain.pl***
    - » **At that time, chainFlag values for previously unflagged chain heads can be set and a chainId can be assigned to each DPR**
    - » **If the Determine Chain script fails for some reason, the plan is not activated**



# Production Requests (Cont.)

- **Intermittent Activation**

- **PGE is set up to run on every *n*th instance of input data**
- **To implement Intermittent Activation the Production Planner supplies the following information (via the Production Request Editor) when creating a production request:**
  - » **“Skip” field on the Production Request Editor: Number of DPRs to be skipped (not executed) is entered in the field**
  - » **“Keep” field on the Production Request Editor: Number of DPRs to keep (after skipping the specified number of DPRs) is entered in the field**
  - » **“Skip First” button on the Production Request Editor: Selected to skip the first DPR (not selected if the first DPR is to be run)**



# Production Requests (Cont.)

- **Intermittent Activation (Cont.)**
  - **Planning Subsystem uses the “Intermittent DPR” information from the PR to establish a pattern of execution**
  - **Pattern is effective for the single PR in which the “number to skip” and the “number to keep” are specified**
    - » **Pattern is not maintained between PRs**

# Production Requests (Cont.)



- **Illustration of Intermittent Activation**
  - **Production Planner prepares a production request for a 14-day period, generating 14 DPRs**
  - **Production Planner made the following selections on the Production Request Editor:**
    - » **Entered “4” in the Number to Skip field**
    - » **Entered “1” in the Number to Keep field**
    - » **Did not select the Skip First button**

# Production Requests (Cont.)



- **Illustration of Intermittent Activation (Cont.)**
  - **The following results are obtained:**
    - » **First DPR runs**
    - » **Four DPRs (second through fifth) are skipped**
    - » **Sixth DPR runs**
    - » **Four DPRs (seventh through tenth) are skipped**
    - » **Eleventh DPR runs**
    - » **Remaining three DPRs (twelfth through fourteenth) are skipped**

# Example of Intermittent Execution



*Run PGE on same data set every five days*

**DAY 1:**



**DAY 6:**



**DAY 11:**



# Production Requests (Cont.)



- **Data Preprocessing (DPREP)**
  - **DPREP (data preprocessing) consists of sets of PGEs that use a statistical approach to convert Level 0 (L0) ephemeris and attitude ancillary data for a particular satellite (e.g., Terra, Aqua) into SDP Toolkit native binary format without altering or modifying the scientific content of the granules**
  - **DPREP PGEs are supplied by ECS, unlike most PGEs**
    - » **Most PGEs are provided by the Science Computing Facilities that ECS supports**
  - **DPREP supports Terra and Aqua operations**

# Production Requests (Cont.)

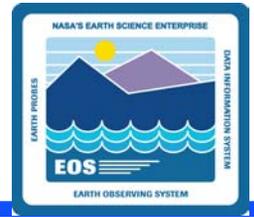


- **Terra DPREP**
  - **Three PGEs:**
    - » **AM1Eph or Step 1**  
**(EcDpPrAm1EdosEphAttDPREP\_PGE)**
    - » **FddAtt or Step 2**  
**(EcDpPrAm1FddAttitudeDPREP\_PGE)**
    - » **RepEph or Step 3**  
**(EcDpPrAm1FddEphemerisDPREP\_PGE)**
  - **Operationally, Steps 1 and 2 are scheduled daily and run independently of one another**
  - **Step 3 is scheduled and run on an as-needed basis**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**
  - **Sources of information on the Terra DPREP PGEs and how to run them:**
    - » **184-TP-001-003, Terra Spacecraft Ephemeris and Attitude Data Preprocessing**
    - » **611-CD-610-002, Mission Operation Procedures for the ECS Project, Chapter 26**
    - » **Two files installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory; i.e., “AM1\_DPREP\_README” and “HowToRunAm1DPREP”**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**

- The DPREP PGEs process Level Zero (L0) Terra (AM-1) spacecraft data (e.g., ESDT AM1ANC) provided by EDOS
- The output files/granules of the DPREP PGEs are subsequently used in the processing of data from various instruments on the satellite. They provide the following types of ancillary (non-science) data:
  - » **Ephemeris - Spacecraft location: ephemeris (or orbit) data include: latitude, longitude, and height**
  - » **Attitude - Orientation of the satellite, including yaw, pitch, and roll angles; and angular rates about three axes**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**
  - **AM1Eph (Step 1 DPREP: EDOS Level 0 Ancillary Data)**
  - **EcDpPrAm1EdosEphAttDPREP\_PGE consists of a script that coordinates the following three DPREP executables:**
    - » **EcDpPrAm1EdosAncillary**
    - » **EcDpPrAm1EdosEphemerisRepair**
    - » **EcDpPrAm1ToolkitToHdf**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**

- **AM1Eph (Step 1 DPREP: EDOS Level 0 Ancillary Data) (Cont.)**

- » **EcDpPrAm1EdosAncillary reads in two granules of AM-1 L0 (EDOS) ancillary data (ESDT AM1ANC) (the second granule ensures that incomplete orbits in the first granule get complete orbit metadata records)**
    - » **EcDpPrAm1EdosAncillary reads in ephemeris and attitude data (ESDT AM1EPHN0 and ESDT AM1ATTN0) (the last ephemeris/attitude granules generated from a previous run of the PGE)**
    - » **If EcDpPrAm1EdosAncillary detects a short gap in the data, EcDpPrAm1EdosEphemerisRepair fills the gap and writes a gap-filled native-format (Toolkit-format) ephemeris file**
    - » **EcDpPrAm1ToolkitToHdf takes the native format ephemeris file and produces a corresponding HDF file and a metadata file**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**
  - **AM1Eph (Step 1 DPREP: EDOS Level 0 Ancillary Data) (Cont.)**
    - » **AM1Eph produces attitude (ESDTs AM1ATTN0 and AM1ATTH0) and ephemeris (ESDTs AM1EPHN0 and AM1EPHH0) granules**
  - **EDOS-supplied ephemeris data are the primary source of ephemeris for Terra**
  - **EDOS-supplied attitude data are **not** the primary source of attitude for Terra**
    - » **Attitude data supplied by the Flight Dynamics Division (FDD) are considered the primary source of Terra attitude**

# Production Requests (Cont.)



- **Terra DPREP (Cont.)**

- **AM1Eph (Step 1 DPREP: EDOS Level 0 Ancillary Data) (Cont.)**
- **EcDpPrAm1EdosAncillary performs a full complement of data quality analyses on the EDOS ephemeris data**
  - » **In contrast EDOS-supplied attitude data are subject to minimal quality checks and never undergo data repair because EDOS-supplied attitude data are not considered good enough for science data processing**
  - » **EDOS attitude data are “use at own risk” data**
  - » **The data recommended for science data processing are the FDD attitude data routinely preprocessed by Step 2 DPREP (EcDpPrAm1FddAttitudeDPREP\_PGE)**

# Production Requests (Cont.)



- **Terra DPREP (Cont.)**
  - **FddAtt (Step 2 DPREP: Attitude Data)**
    - » Reads in both the current FDD attitude granule (AM1ATTF) and the next FDD attitude granule
    - » Reads in the attitude granule (AM1ATTNF) it produced with its last run
    - » Output is native format attitude file (AM1ATTNF), HDF format attitude file (AM1ATTHF) and associated metadata files



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**
  - **RepEph (Step 3 DPREP: Repaired Ephemeris Data)**
    - » If Step 1 finds too many missing data points in the ephemeris data (e.g., AM1EPHH0 and AM1EPHN0 granules have gaps of greater than 60 seconds), it signals that there are problems with the data
    - » Production Planner receives an e-mail message indicating the problems in the preprocessed Terra ephemeris granules and notifies the FDD (e.g., by telephone) that a repaired ephemeris file (AM1EPHF) is needed for the time span of the granule that has the gap
    - » RepEph (Step 3) reads in the definitive ephemeris granule received from FDD and the EOS\_AM1 ephemeris data and produces replacement ephemeris granules (AM1EPHH0 and AM1EPHN0)

# Production Requests (Cont.)



- **Terra DPREP (Cont.)**
  - **DPREP processing has data requirements beyond the current two-hour segment (data from the preceding and following segments are used for performing consistency checks)**
    - » **There is no guarantee that data from the preceding and following segments will always be available**
    - » **Consequently, four data processing profiles have been developed for each of the three DPREP steps to accommodate the various permutations of data availability**



# Production Requests (Cont.)

- **Terra DPREP (Cont.)**

- **Profile 1 (preceding and following data available) is used for nominal DPREP operation**
  - » **It is the profile of each DPREP step that is run on a routine basis**
- **Profile 2 (no preceding data, but following data is available) initializes DPREP's processing of a given step's ephemeris and/or attitude data stream**
  - » **When Profile 2 has been run on a data segment, Profile 1 assumes processing responsibility on all data segments thereafter until data dropout**
- **Profile 3 (preceding data available, but no following data) processes the data segment that immediately precedes data dropout (terminates processing on a given step's ephemeris and/or attitude data stream)**
- **Profile 4 is used for processing isolated data segments (not likely to be scheduled operationally)**



# Production Requests (Cont.)

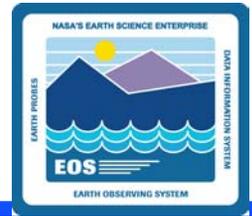
- **Terra DPREP (Cont.)**

- In the big picture of the mission, DPREP processing on the very first data segment would use **Profile 2**
- The next data segments would be processed using **Profile 1** processes
- The very last data segment of the mission could be processed using **Profile 3**
  - » However, PDPS never halts DPREP processing if a following granule does not become available
  - » So if a following granule does not become available after a four-hour wait period, DPREP (Profile 1) reverts to Profile 3 processing, which does not depend on the raw input granule from the segment that follows
- Given the processing flexibility of Profile 1, neither the scheduling of Profile 3 or the scheduling of Profile 4 is envisioned operationally



# Production Requests (Cont.)

- **Aqua DPREP**
  - **Aqua DPREP processes:**
    - » **PM1DefEph (EcDpPrPm1FddEphemerisDPREP\_PGE)**  
**Aqua Ephemeris Processing**
    - » **PM1DefAtt (EcDpPrPm1AttitudeDPREP\_PGE)**  
**Aqua Attitude Processing**
  - **Sources of information on the Aqua DPREP PGEs and how to run them:**
    - » **184-TP-003-002, Aqua Spacecraft Ephemeris and Attitude Data Preprocessing**
    - » **611-CD-610-002, Mission Operation Procedures for the ECS Project, Chapter 26**
    - » **File installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory; i.e., “HowToRunPm1DPREP”**



# Production Requests (Cont.)

- **Aqua DPREP (Cont.)**
  - Includes the processing of both ephemeris and attitude data
  - Ephemeris data
    - » Is received from the FDD in ephemeris data files
    - » Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day
  - Attitude data
    - » Ground-Based Attitude Determination (GBAD) data is used in the processing of attitude data
    - » GBAD data is received from the ECS Mission Operations Segment (EMOS) in "carry-out" files
    - » GBAD carry-out files (PMCOGBAD) are used in conjunction with ephemeris data to prepare attitude data

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **Ephemeris and Attitude PGEs are scheduled daily and run independently of one another**
  - **However, Ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing**



# Production Requests (Cont.)

- **Aqua DPREP (Cont.)**
  - **PM1DefEph (Aqua Ephemeris Processing)**
    - » Includes reformatting the FDD ephemeris granules into Toolkit native format and HDF format
    - » In addition, ephemeris metadata records are generated for the product granules
    - » Aqua Ephemeris DPREP PGE processes Aqua ancillary granules (i.e., PM1EPHD, ephemeris data) supplied by FDD
    - » Output files/granules of the Aqua Ephemeris DPREP PGE provide satellite ephemeris data and are subsequently used in the processing of Aqua satellite attitude data

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **PM1DefEph (Aqua Ephemeris Processing) (Cont.)**
    - » **Aqua Ephemeris DPREP PGE reads in both the current PM1EPHD ephemeris granule and a previous PM1EPHND preprocessed platform ephemeris granule in native format**
    - » **Outputs are preprocessed Aqua platform ephemeris granules in native format (PM1EPHND) and HDF format (PM1EPHHD) and a metadata file for each output data file**

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **PM1DefAtt (Aqua Attitude Processing)**
    - » **Inputs are from ESDTs PMCOGBAD and PM1EPHND (preprocessed Aqua platform ephemeris data in native format)**
    - » **Outputs are PM1ATTHR (Aqua attitude data in HDF-EOS format) and PM1ATTNR (Aqua attitude data in native format)**
    - » **The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals**
    - » **The PMCOGBAD data in the form of two-hour granules (12 granules per day)**
    - » **Each PM1EPHND granule represents 12 hours of data (two granules per day)**

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **Aqua Attitude Processing (Cont.)**
    - » Interpretation of the attitude data depends on the value of the Guidance, Navigation, and Control (GN&C) Status Word 2 contained in the carry-out file GBAD data
    - » In general if the value of GN&C Status Word 2 represents "fine point" mode, the attitude is acceptable for Aqua science processing
    - » Any other value indicates inadequate attitude accuracy for Aqua science data processing

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **Aqua Attitude Processing (Cont.)**
    - » **EcDpPrPm1AttitudeDPREP\_PGE reads in the previous, current, and next FDD attitude granules (PMCOGBAD)**
    - » **It reads in the ephemeris (PM1EPHND) granule for the same time period (and adjacent granule if near a granule boundary)**
    - » **It also reads in the attitude granule (PM1ATTNR) produced during the previous run**
    - » **The output of the process is a native-format attitude file (PM1ATTNR) and an HDF-format attitude file (PM1ATTHR)**
    - » **A metadata file is produced for each output data file**

# Production Requests (Cont.)



- **Aqua DPREP (Cont.)**
  - **Aqua Data Repair**
    - » **Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed**

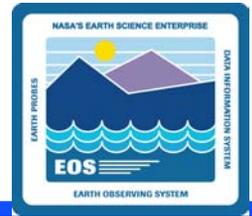


# Production Requests (Cont.)

- **Aqua DPREP (Cont.)**

- **Aqua DPREP Profiles**

- » **As mentioned with regard to Terra DPREP processing Aqua DPREP data requirements extend beyond the current two-hour segment**
    - » **Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries**
    - » **However, there is no guarantee that data from the preceding and following segments will always be available**
    - » **Consequently, four data processing profiles have been developed for each of the Aqua DPREP PGEs to accommodate the various permutations of data availability**
    - » **The Aqua DPREP profiles have the same functions as the corresponding Terra DPREP profiles**



# Production Requests (Cont.)

- **Aura DPREP**
  - **Aura DPREP is very similar to Aqua DPREP**
  - **Aura DPREP processes:**
    - » **AuraEph (EcDpPrAuraEphemerisDPREP\_PGE) Aura Ephemeris Processing**
    - » **AuraAtt (EcDpPrAuraAttitudeDPREP\_PGE) Aura Attitude Processing**
  - **Sources of information on the Aqua DPREP PGEs and how to run them:**
    - » **184-TP-004-001, Aura Spacecraft Ephemeris and Attitude Data Preprocessing**
    - » **611-CD-610-002, Mission Operation Procedures for the ECS Project, Chapter 26**
    - » **File installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory; i.e., “HowToRunAuraDPREP”**



# Production Requests (Cont.)

- **Aura DPREP (Cont.)**
  - **Includes the processing of both ephemeris and attitude data**
  - **Ephemeris data**
    - » **Is received from the FDD in ephemeris data files**
    - » **Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day**
  - **Attitude data**
    - » **Ground-Based Attitude Determination (GBAD) data is used in the processing of attitude data**
    - » **GBAD data is received from the ECS Mission Operations Segment (EMOS) in "carry-out" files**
    - » **GBAD carry-out files (AUCOGBAD) are used in conjunction with ephemeris data to prepare attitude data**

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **Ephemeris and Attitude PGEs are scheduled daily and run independently of one another**
  - **However, Ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing**

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **AuraEph (Aura Ephemeris Processing)**
    - » Includes reformatting the FDD ephemeris granules into Toolkit native format and HDF format
    - » In addition, ephemeris metadata records are generated for the product granules
    - » **Aura Ephemeris DPREP PGE processes Aura ancillary granules (i.e., AUREPHMF, ephemeris data) supplied by FDD**
    - » **Output files/granules of the Aura Ephemeris DPREP PGE provide satellite ephemeris data and are subsequently used in the processing of Aura satellite attitude data**

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **AuraEph (Aura Ephemeris Processing) (Cont.)**
    - » **Aura Ephemeris DPREP PGE reads in both the current AUREPHMF ephemeris granule and a previous AUREPHMN preprocessed platform ephemeris granule in native format**
    - » **Outputs are preprocessed Aura platform ephemeris granules in native format (AUREPHMN) and HDF format (AUREPHMH) and a metadata file for each output data file**



# Production Requests (Cont.)

- **Aura DPREP (Cont.)**
  - **AuraAtt (Aura Attitude Processing)**
    - » **Inputs are from ESDTs AUCOGBAD and AUREPHMN (preprocessed Aura platform ephemeris data in native format)**
    - » **Outputs are AURATTH (Aura attitude data in HDF-EOS format) and AURATTN (Aura attitude data in native format)**
    - » **The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals**
    - » **AUCOGBAD data is in the form of two-hour granules (12 granules per day)**
    - » **Each AUREPHMN granule represents 12 hours of data (two granules per day)**

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **AuraAtt (Aura Attitude Processing) (Cont.)**
    - » Interpretation of the attitude data depends on the value of the GN&C Status Word 2 contained in the carry-out file GBAD data
    - » In general if the value of GN&C Status Word 2 represents "fine point," "attitude hold," or "earth point" mode, the attitude is acceptable for Aura science processing
    - » Any other value indicates inadequate attitude accuracy for Aura science data processing

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **AuraAtt (Aura Attitude Processing) (Cont.)**
    - » **EcDpPrAuraAttitudeDPREP\_PGE reads in the previous, current, and next FDD attitude granules (AUCOGBAD)**
    - » **It reads in the ephemeris (AUREPHMN) granule for the same time period (and adjacent granule if near a granule boundary)**
    - » **It also reads in the attitude granule (AURATTN) produced during the previous run**
    - » **Output of the process is a native-format attitude file (AURATTN) and an HDF-format attitude file (AURATTH)**
    - » **A metadata file is produced for each output data file**

# Production Requests (Cont.)



- **Aura DPREP (Cont.)**
  - **Aura Data Repair**
    - » **Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed**



# Production Requests (Cont.)

- **Aura DPREP (Cont.)**

- **Aura DPREP Profiles**

- » **As mentioned with regard to Terra and Aqua DPREP processing Aura DPREP data requirements extend beyond the current two-hour segment**
    - » **Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries**
    - » **However, there is no guarantee that data from the preceding and following segments will always be available**
    - » **Consequently, four data processing profiles have been developed for each of the Aura DPREP PGEs to accommodate the various permutations of data availability**
    - » **The Aura DPREP profiles have the same functions as the corresponding Terra DPREP profiles**

# Production Requests (Cont.)



- **ASTER Expedited Data Processing**
  - ECS supports the ingest of ASTER expedited granules (EDSs) from the EOS Data and Operations System (EDOS) and their immediate availability to selected ASTER Scientists
  - In addition, ECS supports the production of ASTER Level 1A/1B expedited data products from the ingested expedited granules
  - The higher-level expedited data products are produced at the Land Processes (LP) DAAC



# Production Requests (Cont.)

- **Process for ASTER expedited data processing**
  - **ASTER EDS data granules are transferred to the LP DAAC from the Goddard Space Flight Center (GSFC) Earth Sciences (GES) DAAC via cross-DAAC transfer mechanism**
  - **ASTER Ground Data System (GDS) produces a daily Observation Schedule File (OSF)**
    - » **Including the ASTER Observation Schedule and the ASTER One-Day Schedule**
  - **OSF file is sent daily to the Earth Resources Observation Systems (EROS) Data Center (EDC) (outside the context of ECS at the DAAC)**
  - **OSF is processed by the ASTER OSF Parser tool at LP DAAC (outside the ECS context)**
    - » **Output is referred to as Parsed OSF (POSF) data files**



# Production Requests (Cont.)

- **Process for ASTER expedited data processing (Cont.)**
  - At LP DAAC the POSF files are inserted into the ECS system via the Science Investigator-Led Processing Systems (SIPS) interface mechanism
  - Production Planner deletes all successfully completed L1AE and L1BE DPRs from PDPS
  - Production Planner runs the ASTER Expedited Planning Script (i.e., "GetGranTimes.pl")
    - » Script queries the SDSRV for the temporal metadata of any AST\_EXP and AST\_L1AE granules that were inserted to SDSRV since the last time it was run
    - » Output is a set of candidate DPR start/stop times



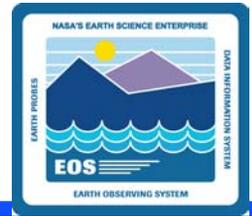
# Production Requests (Cont.)

- **Process for ASTER expedited data processing (Cont.)**
  - **Production Planner generates a set of Production Requests (PRs) for both the L1AE PGE and the L1BE PGE using the DPR start/stop times identified using the GetGranTimes.pl script**
    - » **PGEs are configured to run with Collection Time (rather than Insertion Time) production rules**
  - **Production Planner generates PRs (Cont.)**
    - » **If the matching AST\_POSF granule has not been inserted, a L1AE PR is still generated with a subscription for AST\_POSF**
    - » **L1BE PGEs being planned are not those corresponding to the L1AE PGEs being planned; L1AE PGEs must be finished and their DPRs deleted from PDPS prior to planning the corresponding L1BE PGEs**



# Production Requests (Cont.)

- **Process for ASTER expedited data processing (Cont.)**
  - Production Planner creates a new production plan, merging the newly created L1AE and L1BE PRs with any PRs from the currently active plan
  - Production Planner activates the new plan for execution of L1AE and L1BE DPRs
  - Subsequent processing, archiving and access activities are the same as for other data types



# Production Requests (Cont.)

- **ASTER On-Demand Processing**
  - ECS supports requests for on-demand high-level production
  - Requests can be divided into the following categories:
    - » Pre-defined (standard) ASTER high-level products
    - » High-level products with nonstandard input parameters
    - » Digital Elevation Model (DEM) products
  - Requester selects the granule to be used and/or the additional non-standard parameters to generate the high-level product and submits a request for the system to create the high-level product
  - Granules produced by on-demand processing are not permanently archived



# Production Requests (Cont.)

- **ASTER On-Demand Processing: Standard or Non-Standard On-Demand Processing**
  - Supports requests for processing of ASTER L1A and L1B data to higher information levels
  - A request may require a sequence of algorithms to be run on the specified data
  - Requester uses the EOS Data Gateway (EDG) web client to submit ASTER On-Demand Product Requests via the ECS V0 Gateway
  - On-Demand Manager (EcPIOdMgr) in PLS manages on-demand orders received from the EDG
    - » Determines the order type
    - » Creates the PRs and DPRs necessary to fill the order
    - » If all input data are available, the DPR(s) is (are) submitted to the Data Processing Subsystem



# Production Requests (Cont.)

- **ASTER On-Demand Processing: Standard or Non-Standard On-Demand Processing (Cont.)**
  - Job Management (EcDpPrJobMgmt) gets activated on-demand DPRs from PLS
  - If the on-demand DPRs are waiting for data to arrive or if there are not enough processing resources, Job Management places them in the On-Demand Queue
  - When all input data are available and there are adequate processing resources, the on-demand DPRs are released to AutoSys for processing
  - Execution Management (EcDpPrEM) sends updates to MSS concerning the status of the on-demand jobs

# Production Requests (Cont.)



- **ASTER On-Demand Processing: DEM Production**
  - Requests for **ASTER on-demand DEM production** are submitted to **ECS** through the **EDG** and are forwarded to the **ASTER DEM SIPS** for processing
    - » The **ASTER DEM SIPS** is located at **EDC** in close proximity to the **LP DAAC**
  - When the **ASTER DEM SIPS** has created the requested product, **ECS** ingests it and distributes the product to the requester



# Production Requests (Cont.)

- **ASTER On-Demand Processing: DEM Production (Cont.)**
  - The ASTER scientist submits a request for ASTER on-demand DEM production to ECS through the EDG
  - Upon receipt of an on-demand order, ODPRM determines the order type
  - The ODPRM submits a subscription (to the subscription server) to be notified when the ASTER DEM product has been inserted in the archive
  - The ODPRM sends an e-mail message to the ASTER DEM Operator
  - The ASTER DEM Operator creates the DEM in accordance with the order information



# Production Requests (Cont.)

- **ASTER On-Demand Processing: DEM Production (Cont.)**
  - **When the requested product has been created, the ASTER DEM operator puts a PDR file on the PDR server**
    - » **The PDR file identifies the location of the ASTER DEM data file and the corresponding metadata file**
  - **ECS ingests the DEM product using the polling-with-delivery-record interface**
  - **Upon successful insertion of the DEM product in the archive, the science data server triggers the appropriate "insert event" qualified with the universal reference (UR) of the inserted granule**
  - **The subscription server queries the subscription database to determine which subscriptions need to be activated (fired)**



# Production Requests (Cont.)

- **ASTER On-Demand Processing: DEM Production (Cont.)**
  - The status of the on-demand request is updated to “Processing Complete”
  - ODPRM receives subscription notification and sends a "user acquire" request to the science data server to have the on-demand product sent to the requester
  - The DEM product is sent to the requester via the requested medium
    - » The data may be sent via ftp or on a physical medium (e.g., tape)

# Production Requests (Cont.)



- **Reprocessing**

- **"Reprocessing" has been used in the following two ways in the ECS context:**
  - » **A PGE has been run in the past and produced a product and for some reason (e.g., a file has been corrupted/deleted) the PGE needs to be run again to reproduce the exact same outputs**
  - » **A PGE that was run in the past over a given time period or set of data has been improved (software or static inputs) by the Instrument Team, who would like to have the modified PGE run on the data from the previously processed time period/data set**
- **The second case is the more important**
  - » **Leads to the requirement that ECS be able to support a workload for reprocessing that is equal in size to the current or first-time processing load**



# Production Requests (Cont.)

- **Reprocessing (Cont.)**
  - **PGE changes leading to reprocessing may occur for any of the following reasons:**
    - » **Correct an error in the software**
    - » **Improve the algorithm on the basis of better understanding of the instrument or physical phenomena**
    - » **Use updated static files (e.g., calibration data)**
    - » **Use new or different ancillary data files**
    - » **Use different production rules**
    - » **Use changed/additional run-time parameters**
    - » **Use improved input resulting from changes to an upstream PGE**



# Production Requests (Cont.)

- **Reprocessing:  
Operations Concepts**
  - A PGE that is being used for reprocessing represents a new version of a PGE
  - A reprocessing PGE may specify new static or dynamic inputs rather than new PGE software
  - Output granules from reprocessing PGEs are represented under the existing ESDT used for the previous generation or version of the data granules
  - The distinction between generations of the data can be made through the `ReprocessingActual` attribute that is associated with each granule
    - » Other mechanisms for distinguishing reprocessing levels include the `ProductionDateTime` and `PGEVersion` attributes



# Production Requests (Cont.)

- **Reprocessing:  
Operations Concepts (Cont.)**
  - DPRs specifying the exact same PGE may be scheduled for both reprocessing and routine processing in the same production plan
    - For example, first-time processing may have been carried out for the time period from Day 1 to Day N, at which point an improved PGE is introduced
    - The improved PGE is used for processing from Day N+1 forward, and is also used concurrently in reprocessing from Day 1 to Day N
  - It is expected that in a typical planning session the Production Planner will plan time ranges for reprocessing that are comparable in extent with the routine processing
    - » For example, the Production Planner may plan both routine processing and reprocessing for data acquired during 24-hour periods



# Production Requests (Cont.)

- **Reprocessing:  
Operations Concepts (Cont.)**
  - DPRs created via planning are organized and managed in separate queues by processing type
    - » A fixed number of processing "slots" are reserved for on-demand processing, routine processing, and reprocessing
    - » DPRs are dispatched for processing from either queue against the appropriate slot when the slot becomes available, i.e., as a job of that slot's type finishes and the input data become available
    - » Operations personnel can configure the number of slots of each type
    - » Operations personnel can change the number of slots of each type (and as a result the total number of concurrent processing slots) at run time
    - » There is no dynamic slot allocation (allocation is done by Operations personnel)



# Production Requests (Cont.)

- **Regeneration**

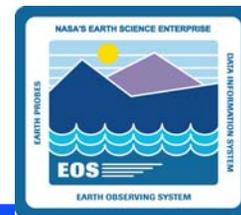
- **Produce replacements for previously generated granules that have been lost or corrupted due to failure in the ECS archive**
- **General Process:**
  - » **Retrieve the Production History file (PH) for the lost granule to determine parameters for the generation of replacement granules**
  - » **Create Production Requests for the generation of replacement granules**
  - » **Create and activate a Production Plan that includes the Production Requests for the generation of replacement granules**
  - » **Prepare (if applicable) a “PDPS Residual Granules List,” which identifies granules that either cannot or should not be regenerated at the DAAC**
  - » **Some granules do need not be reproduced; e.g., if there is a more recent version of the product**

# Regenerating Granules (Cont.)



- **Considerations that apply to the regeneration of granules:**
  - **All outputs of the PGE [not just those equivalent to the lost granule(s)] are to be produced and archived**
  - **Various factors could make it impossible to reproduce granules identical to the originals**
    - » **There is no guarantee that when a PGE is re-run it will use the same inputs as were used during the original execution of the PGE; consequently, the output may be different from the original granule(s)**
    - » **Variability of Optional/Alternate inputs, Ad Hoc Reprocessing, Metadata Checks, Metadata Query and other production rules affects PGE output**

# Logging in to ECS Hosts



- **Logging in to ECS hosts is accomplished from a UNIX command line prompt**
  - It is an initial set of steps that is performed when accomplishing many other Production Planning and Processing tasks
- **Procedure**
  - Access the command shell
  - Set the **DISPLAY** environmental variable
  - Log in to the specified host using secure shell and the specified user ID

# Launching the Production Request Editor



- **Production Request Editor-Associated Applications**
  - Production Request Editor
  - Subscription Manager
  - Sybase ASE Server for the PDPS database

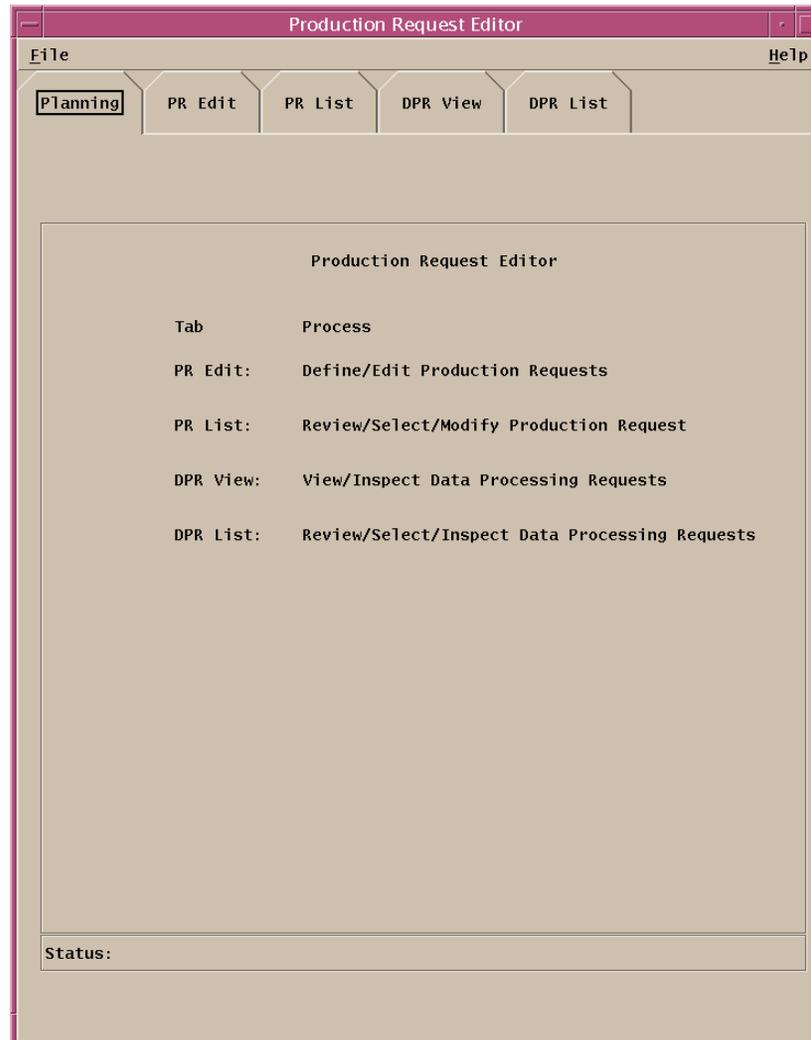
# Launching the Production Request Editor (Cont.)



- **Procedure**

- **Access a terminal window logged in to the Planning/Management Workstation host**
- **Set the ECS\_HOME environmental variable if necessary**
- **Change directory to the subdirectory (e.g., utilities) containing the production planning startup scripts**
- **Start the Production Request Editor GUI in the appropriate mode**

# PR Editor Introductory GUI



# Creating a New PR Using the Production Request Editor GUI



- **Procedure**

- **Select the PR Edit tab of the Production Request Editor GUI**
- **Select the PGE to be included in the Production Request**
- **If applicable, specify new metadata check value**
- **If applicable, modify alternate input parameters**
- **Specify the data requirements**
  - » **Beginning and ending dates and times**
  - » **Orbits (from/to)**
  - » **Tile ID**
- **Indicate whether the DPRs will be chain heads**
- **If applicable, select the virtual computer for running the PGE**
- **Save the Production Request**

# PR Edit GUI



Production Request Editor

File Edit Help

Planning PR Edit PR List DPR View DPR List

PR Name:  Origination Date:   
PR Type:  Originator:   
User Type:  Priority (1 to 100):

Satellite Name:  PGE ...  
Instrument Name:  PGE Parameters...  
PGE Name:  Metadata Checks...  
PGE Version:  Alternate Input Values..  
Profile Id:

Collection Time  Insertion Time

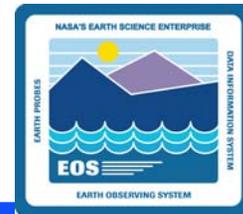
Duration  UTC Time  Orbit

Collection Time  
Begin  From   
End  To   
Tile Id

PGE Chain Head  Yes  No Computer

Intermittent DPR  
Skip  Keep   SkipFirst

# PR Edit GUI with Multiple DPRs Toggle Button



Production Request Editor

File Edit Help

Planning PR Edit PR List DPR View DPR List

PR Name:  Origination Date:   
PR Type:  Originator:   
User Type:  Priority (1 to 100):

Satellite Name:  PGE ...  
Instrument Name:  PGE Parameters...  
PGE Name:  Metadata Checks...  
PGE Version:  Alternate Input Values...  
Profile Id:

Collection Time  Insertion Time

Duration  Multiple DPRs

Insertion Time (Local Time)

Begin  /  /  -  :  :

End  /  /  -  :  :

PGE Chain Head

Yes  No

Intermittent DPR

Skip  Keep   SkipFirst

# PGE Selection GUI



PGE Selection

PGE Name	PGE Version	Profile Id	Satellite Name	Instrument Name
ACT	OnDemand	1	AM1	ASTER
ACT	OnDemand	2	AM1	ASTER
ACT	OnDemand	3	AM1	ASTER
ACT	OnDemand	4	AM1	ASTER
ACT	syn1	1	AM1	ASTER
ACVS	OnDemand	1	AM1	ASTER
ACVS	OnDemand	2	AM1	ASTER
ACVS	OnDemand	3	AM1	ASTER
ACVS	OnDemand	4	AM1	ASTER
AM1Eph	30101	1	AM1	AT1
AM1Eph	30101	2	AM1	AT1
BTS	OnDemand	1	AM1	ASTER
BTS	syn1	1	AM1	ASTER
DSTSD	OnDemand	1	AM1	ASTER
DSTTD	OnDemand	1	AM1	ASTER
DSTVD	OnDemand	1	AM1	ASTER
ETS	OnDemand	1	AM1	ASTER
ETS	OnDemand	2	AM1	ASTER
ETS	OnDemand	3	AM1	ASTER
ETS	OnDemand	4	AM1	ASTER
ETS	syn1	1	AM1	ASTER
FddAtt	30101	1	AM1	AT1
FddAtt	30101	2	AM1	AT1
MISRLIKE	syn1	1	AM1	MISR
MISRLIKE	syn1	2	AM1	MISR
MODPGE01	syn1	1	AM1	MODIS
MODPGE02	syn1	1	EOSAM1	MODIS
MoPGE01	syn1	1	AM1	MODIS
MoPGE02	syn1	1	EOSAM1	MODIS
MoPGE03	syn1	1	EOSAM1	MODIS

Find

OK Cancel Help

# PGE Parameter Mappings GUI



PGE Parameter Mappings

PGE ID:

Parameter Name	Logical Id	Def
Spacecraft Name	950	EOS
AlternateInputOne	970	0
AlternateInputTwo	980	0
PGE Version	990	3010
Profile ID	999	1
Toolkit Format Attitude Data Logical ID	1255	100
Toolkit Format Ephemeris Data Logical ID	1256	115
Maximum QA Check Window Size (number of records)	5000	75
Minimum QA Check Window Size (number of records)	5001	23
Position Vector Yellow Limit (st dev or absolute)	5004	8.6
Position Vector Red Limit (st dev or absolute)	5005	15.5
Velocity Vector Yellow Limit (st dev or absolute)	5006	8.6
Velocity Vector Red Limit (st dev or absolute)	5007	15.5
Ephemeris Long Gap Threshold (number of records)	5008	60
Attitude Long Gap Threshold (number of records)	5009	1
Limits Threshold Unit; 0 = st dev, 1 = absolute	5025	0
Ephemeris QA Uncertainty (st dev)	5026	0.0
Generate Attitude Data; 0 = FALSE, else TRUE	5028	1
Generate Ephemeris Data; 0 = FALSE, else TRUE	5029	1
Maximum S/C Distance from ECI center (meters)	5030	7380
Minimum S/C Distance from ECI center (meters)	5031	6500
Maximum S/C Velocity about ECI center (meters/sec)	5032	8300
Minimum S/C Velocity about ECI center (meters/sec)	5033	6850
Ignore replacement request; 0 = FALSE, else TRUE	5038	0
Euler Angle Order, 1st angle; 1 = X, 2 = Y, 3 = Z	5039	3
Euler Angle Order, 2nd angle; 1 = X, 2 = Y, 3 = Z	5040	1
Euler Angle Order, 3rd angle; 1 = X, 2 = Y, 3 = Z	5041	2
Expected Data Interval (seconds)	5050	1.0
Maximum Orbital Period (seconds)	5053	6420
Minimum Orbital Period (seconds)	5054	5250

Find

Override Value

# Metadata Checks GUI



MetaDataChecks

PGE ID:

InputDataType

- AM1ANC#001
- AM1EPHN#001
- AM1ATTN#001

Find

MetaDataField Operator Value Type

Find

MetaDataField:

Value:

OK    Apply    Cancel    Help

# Alternate Input Values GUI



AlternateInputValues

PGE ID: AM1Eph#30101#001

AlternateListName

ANC
ATTNO
EPHNO

Find

Order

Order	DataType	LogicalID	Timer
1	AM1ANC#001	1010	14400

Order

▲

▼

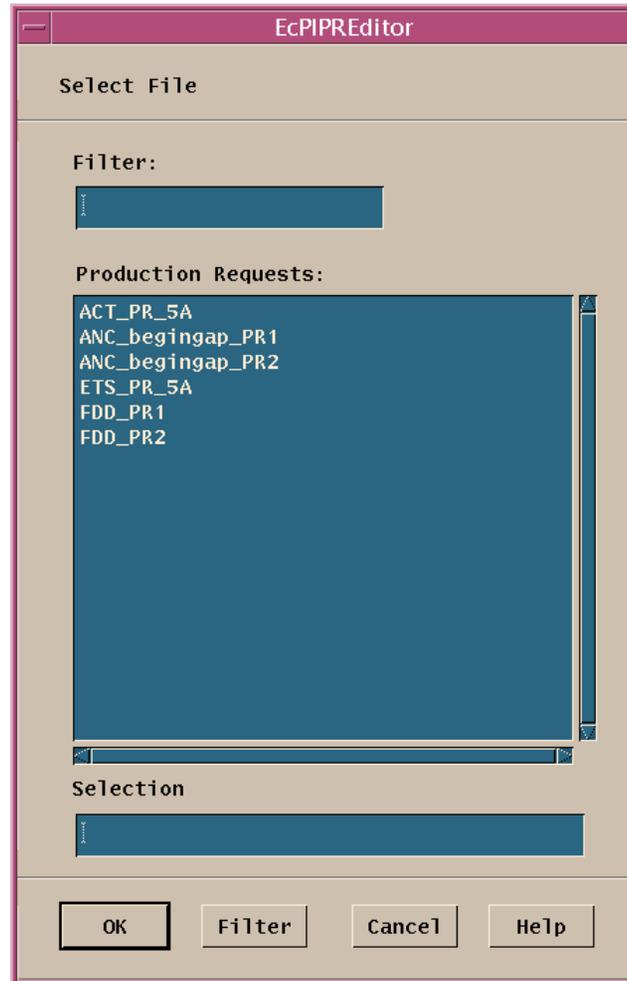
Find

DataType: AM1ANC#001

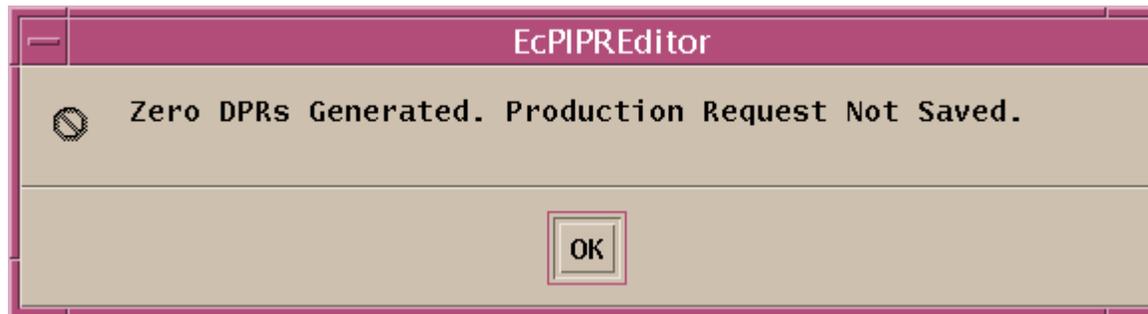
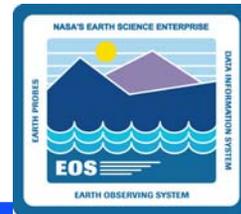
Timer: 04 : 00 : 00

Ok Apply Cancel Help

# PRE File Selection Window ("Open" or "Save As")



# Production Request Explosion into DPRs Dialogue Box



# Creating New PRs Using the Production Request Generator



- **Procedure**

- Log in to the appropriate host
- Prepare an input file specifying the Pgeld and GEOld values to be used in creating the production requests
- Start the PR Generator to create new production requests
- Check the PR Generator debug log to determine the results of running the PR Generator

# Editing/Modifying a PR



- **Procedure**

- **Select the PR Edit tab of the Production Request Editor GUI**
- **Open the PR to be edited/modified**
- **Make the necessary edits/modifications**
- **Save the PR**

# Deleting a PR



- **Procedure**
  - **Select the PR List tab of the Production Request Editor GUI**
  - **Select the PR to be deleted from those listed**
  - **Select Edit→Delete**
  - **Select OK**
  - **Select OK**

# PR List GUI



Production Request Editor

File Edit Help

Planning PR Edit **PR List** DPR View DPR List

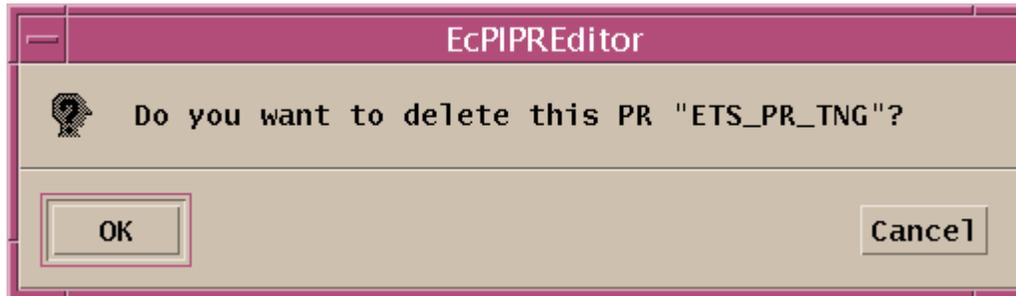
Refresh PR Type: All

Production Requests

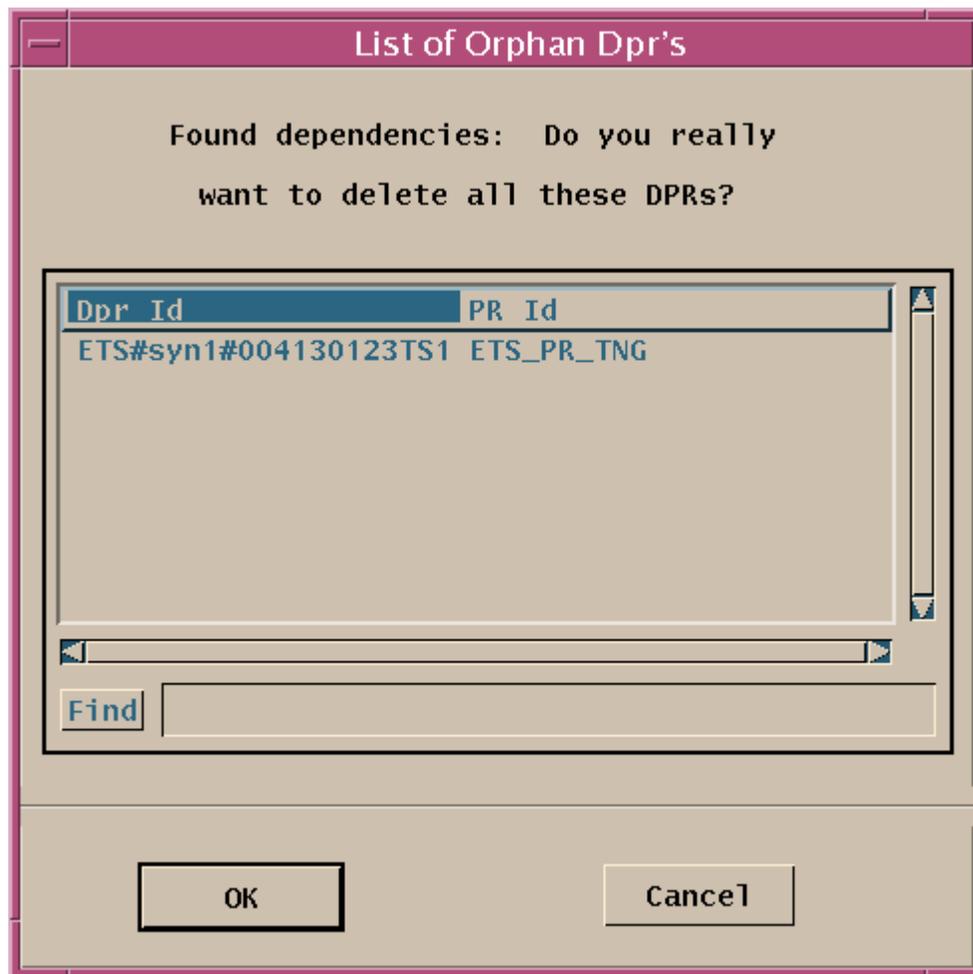
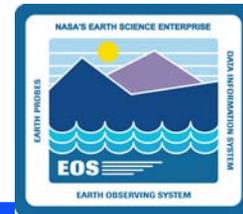
PRName	PGEID	Priority	StartTime	EndTime
A1A_AIRS_330	A1A_AIRS#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1A_AMSU_330	A1A_AMSU#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1A_HSB_330	A1A_HSB#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1B_AIRS_330	A1B_AIRS#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1B_AMSU_330	A1B_AMSU#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1B_HSB_330	A1B_HSB#syn01#001	10	03/30/2000 00:00:00	03/31/2000
A1B_VIS_330	A1B_VIS#syn01#001	10	03/30/2000 00:00:00	03/31/2000
AL2_AIRS_330	AL2_AIRS#syn01#001	10	03/30/2000 00:00:00	03/31/2000
AM1Att_328_329	FddAtt#syn001#001	10	03/28/2000 02:00:00	03/29/2000
AM1Eph_224	AM1Eph#syn001#001	10	02/24/1999 00:00:00	02/24/1999
AM1Eph_328_329	AM1Eph#syn001#001	10	03/28/2000 02:00:00	03/29/2000
FddAtt_224	FddAtt#syn001#001	10	02/24/1999 00:00:00	02/24/1999
MoPGE01_224	MoPGE01#syn001#001	10	02/24/1999 00:00:00	02/24/1999
MoPGE01_328_329	MoPGE01#syn001#001	10	03/28/2000 02:00:00	03/29/2000
MoPGE02_224	MoPGE02#syn001#001	4	02/24/1999 00:00:00	02/24/1999
MoPGE02_328_329	MoPGE02#syn001#001	4	03/28/2000 02:00:00	03/29/2000
MoPGE03_224	MoPGE03#syn001#001	4	02/24/1999 00:00:00	02/24/1999
MoPGE03_328_329	MoPGE03#syn001#001	4	03/28/2000 02:00:00	03/29/2000
MyPGE01_330	MyPGE01#syn001#001	10	03/30/2000 00:00:00	03/30/2000
MyPGE02_330	MyPGE02#syn001#001	50	03/30/2000 00:00:00	03/30/2000
MyPGE03_330	MyPGE03#syn001#001	1	03/30/2000 00:00:00	03/30/2000
PM1Att_330	PM1Att#syn001#001	10	03/30/2000 00:00:00	03/30/2000
PM1EPH_329_330	PM1Eph#syn001#001	10	03/29/2000 00:00:00	03/30/2000

Find

# Production Request Deletion Confirmation Dialogue Box



# Production Request List of Orphan DPRs Dialogue



# Production Request Deletion Completed Dialogue Box

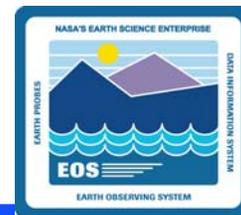


# Reviewing/Deleting Data Processing Requests (DPRs)



- **Data Processing Requests (DPRs)**
  - generated automatically by the PDPS
  - generated automatically from a PR (which specifies a PGE)
- **DPR information is used by...**
  - Data Processing Subsystem
  - AutoSys production scheduling software
- **The Production Planner can review DPRs**
- **Operations personnel cannot edit DPR fields**

# Reviewing/Deleting DPRs (Cont.)



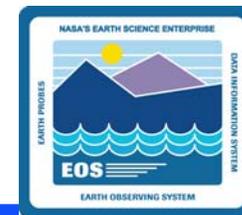
- **DPR List (Production Request Editor)**
  - Each line of the Data Processing Request table represents a DPR, i.e., a job that will be run when all data and resource needs have been satisfied
  - For each DPR the table includes...
    - » DPR identification
    - » relevant PGE
    - » name of the corresponding PR
    - » data start date and time, etc.
  - may be filtered, so that only DPRs with certain characteristics are displayed

# Reviewing/Deleting DPRs (Cont.)



- **Data concerning an individual DPR (Production Request Editor)**
  - **PGE parameters**
  - **UR File Mappings (PGE File Mappings)**
    - » **input and output files for a particular DPR on the PGE File Mappings GUI**
    - » **GUI displays one line of information for each file that may be used by or be produced by the PGE**

# UR File Mappings GUI



UR File Mappings

File Mappings

Input Data

LogicalId	GranuleId	StartTime(UTC)	Stop
1200	AST_ANC#001L1004	01/01/1901 00:00:01	01/01/1901 00:00:01
1100	AST_L1B#00107031997004900000	07/03/1997 00:49:00	07/03/1997 00:49:00
1101	GDAS_OZF#00101003	01/01/1901 00:00:01	01/01/1901 00:00:01

Find

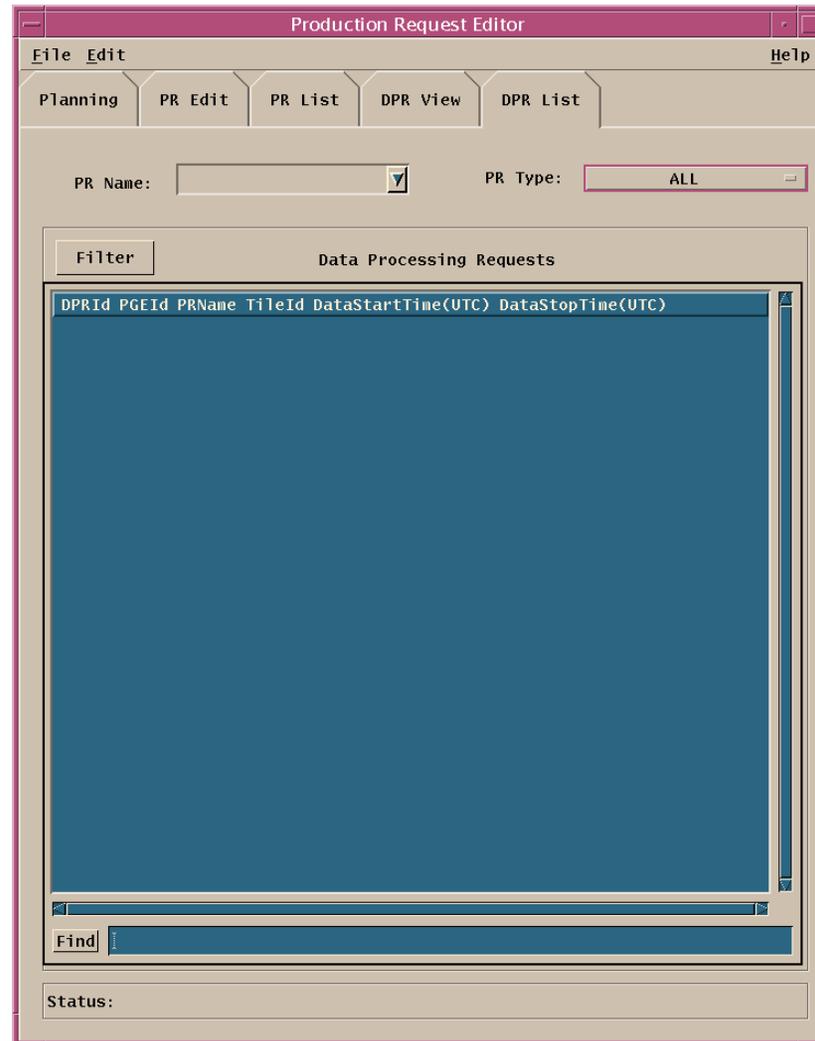
Output Data

LogicalId	GranuleId	StartTime(UTC)	Stop
2000	AST_09T#00107031997004900000	07/03/1997 00:49:00	07/03/1997 00:49:00

Find

OK Help

# DPR List GUI



# Reviewing/Deleting DPRs (Cont.)



- **Procedure (Reviewing DPRs)**
  - **Select the Data Processing Requests list by clicking on the DPR List tab**
  - **Select a Production Request from the list on the PR Name option button**
  - **Click on a DPR to be reviewed from the list of Data Processing Requests**
  - **Select File → Open from the pull-down menu**
  - **Click on the DPR View tab**
  - **Review the selected DPR**
    - » **Click on the PGE Parameters... button to view the PGE parameters associated with the DPR**
    - » **Click on the PGE File Mappings... button to view the UR file mappings (PGE input and output data)**

# DPR View GUI



Production Request Editor

File Edit Help

Planning PR Edit PR List **DPR View** DPR List

All Times In UTC

Data Processing Request Identification

DPR Name:  PR Name:

Origination Date:

Originator:

PGE ID:  PGE Parameters...

Data Start Time:

Data Stop Time:  PGE File Mappings...

Request Data and Status

Predicted Start

Time:  Priority:

Actual Start

Time:  Status:

# Reviewing/Deleting DPRs (Cont.)



- **Procedure (Deleting a DPR)**
  - **Select DPR List tab on the Production Request Editor GUI**
  - **Select the appropriate Production Request from the list on the PR Name option button**
  - **Select the DPR to be deleted from the list of Data Processing Requests**
  - **Select Edit→Delete**
  - **Select OK**
  - **Select OK**

# DPR Deletion Confirmation Dialogue Box



# DPR Deletion Completed Dialogue Box



# Reviewing/Deleting DPRs (Cont.)



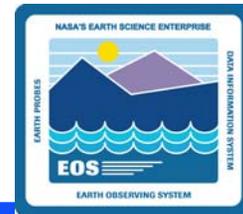
- **Deleting or Aborting an On-Demand Processing Request**
  - **The Planning Subsystem detects changes in the status of ASTER on-demand processing requests**
    - » **Whether the changes were made by the system or by operations personnel**
  - **PLS is not able to determine why operations personnel have canceled or aborted a request**
  - **Consequently, if it is necessary to cancel or abort an on-demand processing request, notify User Services personnel**
    - » **So they can send an e-mail message to the requester explaining why the request was canceled or aborted**

# Launching Planning Workbench-Related GUIs



- **Launching the Production Strategies GUI**
  - **Software applications associated with the Production Strategies GUI**
    - » **Production Strategies GUI**
    - » **Sybase ASE Server for the PDPS database**
  - **Production Strategies are high-level sets of priorities that the Production Planner makes available to the Planning Workbench for determining the priorities and preferences in the processing of DPRs**
  - **Values included in the selected strategy are read by the Planning Workbench when prioritizing the DPRs in a production plan**

# Launching Planning Workbench-Related GUIs (Cont.)



- **Launching the Production Strategies GUI: Procedure**
  - **Access a terminal window logged in to the Planning/Management Workstation host**
  - **Set the ECS\_HOME environmental variable if necessary**
  - **Change directory to the subdirectory (e.g., utilities) containing the production planning startup scripts**
  - **Start the Production Strategies GUI in the appropriate mode**

# Production Strategies GUI



ProductionStrategies

File Edit Options Help

Production Strategies Default

PR Type      User Type      PGE Type

Weight: 50.000000      Weight: 0.000000      Weight: 0.000000

Default: 5      Default: 5      Default: 5

Value	Priority
Routine	5
OnDemand	7
Reprocessing	3

Find

Weight: 50.000000

Total Weight (must total 100): 100.000000

Normalize

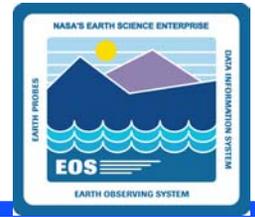
Priority: 1

Type List

Add      Modify      Delete

Late Start Delta Priority: 0

# Launching Planning Workbench-Related GUIs (Cont.)



- **Planning Workbench-Related Applications**
  - **System Name Server**
  - **Message Handler**
  - **Resource Model**
  - **Planning Workbench**
  - **Production Timeline**
  - **Production Strategies**
  - **Sybase ASE Server for the PDPS database**

# Launching Planning Workbench-Related GUIs (Cont.)



- **Launching Planning Workbench-Related GUIs: Procedure**
    - Access a terminal window logged in to the Planning/Management Workstation host
    - Set the `ECS_HOME` environmental variable if necessary
    - Change directory to the subdirectory (e.g., utilities) containing the production planning startup scripts
    - Start the Planning Workbench GUI in the appropriate mode
- NOTE:** Normally, the `EcPIAllStart` script is used to start all processes. But if the number of DPRs in the PDPS database is very high (~4000), the `EcPISomeStart` script is used to start the underlying processes. Then additional scripts are used to start the Planning Workbench GUI and the Timeline GUI.

# Message Handler GUI



```
Message Handler
[Engineering] [Information] [Warning] [Internal] Filter Clients
feb-02 08:29:27 <EcPlRm3-TS2> internal Connected to Message Handler
feb-02 08:29:27 <EcPlRm3-TS2> information (INFO) Loading resources from database
feb-02 08:29:29 <EcPlRm3-TS2> information (INFO) Loading ground events from database
feb-02 08:29:29 <EcPlRm3-TS2> information (INFO) Loading data processing requests from database
feb-02 08:29:31 <EcPlRm3-TS2> information (INFO) Creating new DPRs.
feb-02 08:29:31 <EcPlRm3-TS2> information (INFO) Resolving DPR dependencies.
feb-02 08:29:32 <EcPlRm3-TS2> information (INFO) Extracting PGE information.
feb-02 08:29:33 <EcPlRm3-TS2> information (INFO) Loading production requests from the database
feb-02 08:29:34 <EcPlRm3-TS2> information (INFO) Loading plans from database
feb-02 08:29:35 <EcPlRm3-TS2> information (INFO) Finished loading all records from the database
```

# Planning Workbench



Planning Workbench

File Options Help

Plan Name: orbitTest2\_9  
Status: ACTIVE  
Strategy: Default

Baseline  
Activate Rollover Time:

Comments:

Production Requests

Unscheduled:

NAME	PRIORITY
ACT_PR_TNG	2
ETS_PR_TNG	2

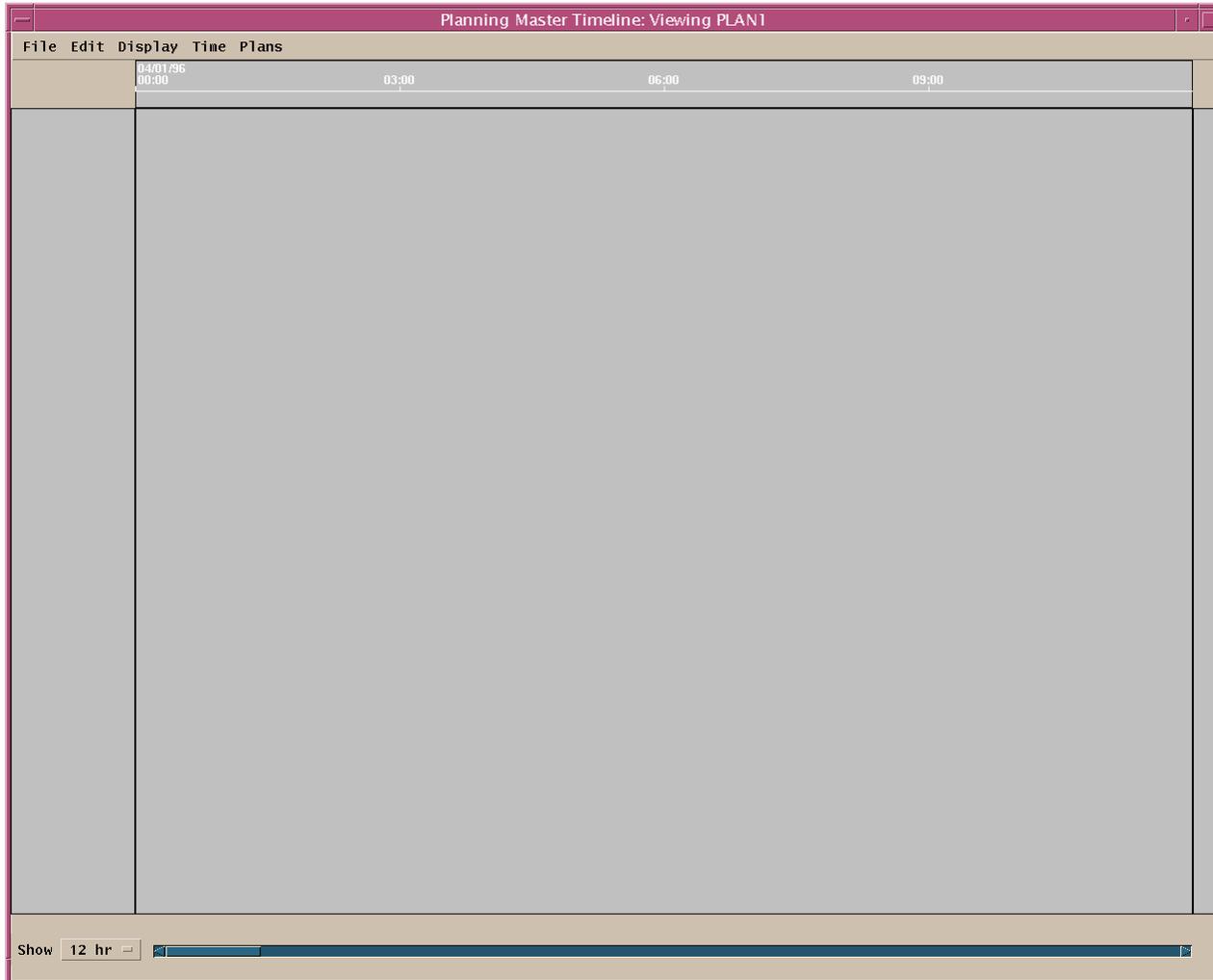
schedule: ▼ unschedule: ▲

Scheduled:

NAME	PRIORITY
ACT_PR1	250

Prioritize Refresh

# Planning Timeline GUI



# Creating a New Production Plan



- **Planning process**
  - Involves the Production Planner preparing monthly and weekly production plans as well as a daily production schedule from the most current weekly plan
    - » Much of the planning occurs off line
    - » However, the Production Planner uses the Planning Workbench GUI to create and activate the current or active plan
- **Each DAAC has created its own policy and process for creating, reviewing, publishing and distributing production plans**
  - Daily plans may be...
    - » posted on a Worldwide Web page
    - » distributed in hardcopy form
    - » distributed via e-mail

# Creating a New Production Plan (Cont.)



- **Most important factor in production planning is meeting the customers' needs**
  - **Ultimate customers are the Science Computing Facilities that have provided science software to ECS to be run on a routine basis**
    - » **They expect timely generation and archiving of products using their science software**
  - **Factors to consider when creating production plans**
    - » **Keeping up to date with routine processing of products using the SCFs' science software**
    - » **Performing reprocessing of data using new versions of science software**
    - » **Regenerating products (as needed) that have been damaged or lost from the archive**
    - » **Taking into account the effects of ASTER on-demand processing, which circumvents the normal planning process but uses processing resources**

# Creating a New Production Plan (Cont.)



- **Monthly plans**
  - developed for the coming month and one or two months in advance
  - produced, reviewed, updated, published and distributed approximately two weeks before the beginning of the month
  - used to establish a baseline against which production targets can be measured
- **Weekly plans**
  - produced, reviewed, updated, published and distributed approximately five days before the beginning of the coming week
  - used to produce a baseline for comparison of planned vs. actual production results

# Creating a New Production Plan (Cont.)



- **Daily plan or schedule**
  - produced each day for the next processing day
  - developed from the current weekly plan
    - » adjusted to reflect the actual processing accomplished and the actual resources available at the time the daily schedule is generated
- **Current (active) plan**
  - activated as needed (one or more plans a day, depending on circumstances) using the Planning Workbench
  - implements the daily plan (or some portion of the daily plan)

# Creating a New Production Plan (Cont.)



- **During normal processing, when reasonably accurate predictions of the processing time for the PGEs are available, the processing schedule should result in a reasonably accurate prediction of when data products will be generated**
- **In abnormal situations (e.g., hardware or software failure), what is actually accomplished could depart significantly from the plan**

# Creating a New Production Plan (Cont.)



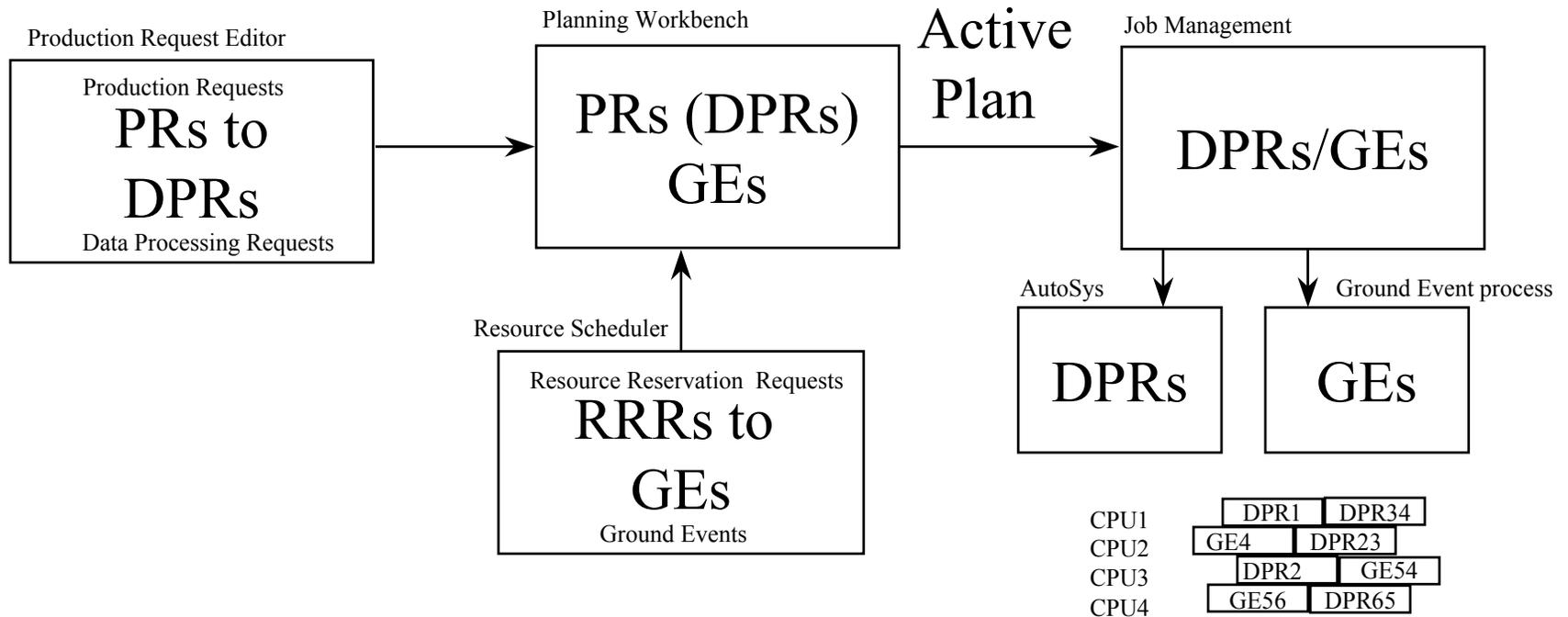
- **A daily plan may be divided into several segments to be activated separately**
  - In such situations, the Production Planner is likely to develop a new current plan to continue processing or to change the processing priorities
  - The process is known as “replanning”
- **Production Planner uses the Planning Workbench when creating a current (active) plan for production data processing at the DAAC**
  - Planning Workbench provides the means by which the Production Planner selects and schedules specific PRs whose DPRs are to be run
  - Planning Master Timeline provides a forecast of the start and completion times of the jobs based upon experience in running the PGEs during SSI&T

# Creating a New Production Plan (Cont.)

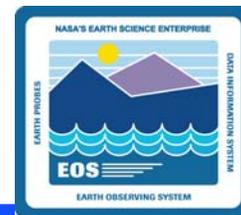


- **When a plan is “activated” through the Planning Workbench, information in the plan is transferred to Job Management in the Data Processing Subsystem**
  - **Job Management creates and releases jobs into the Platinum AutoSys tool, where production processing is managed**

# Creating a New Production Plan: Planning Workbench



# Production Strategy



- **High-level set of priorities created by the Production Planner**
- **Made available for selection on the Planning Workbench**
- **Production Planner's choice of a strategy for a current production plan determines the priorities and preferences in the processing of associated DPRs**

# Production Strategy (Cont.)



- **Production Strategies work on two levels:**
  - Updating lists of DPR attributes so that each value an attribute can have is tied to a particular priority
  - Changing the weight that each attribute's priority is given
- **In addition, weight is given to the priority selected by the user who entered the request**

# Production Strategy (Cont.)



- **Total weights assigned to PR Type, User Type, PGE Type and Production Request Editor [Priority] must equal 100**
- **The values included in a strategy are read by the Planning Workbench to prioritize the DPRs in a plan**
- **Late Start Delta can be used to increase the priority of all jobs that have been waiting in the Production Queue for more than a day**

# Production Strategy: Calculating Priority for a DPR



**DPR id:**  
**MOD09:L2G\_060199\_14\_1234**  
 User Type:  
 Scientist  
 PGE:  
 MOD09:L2G  
 Routine request  
 Tile 14  
 Cluster id: 4  
 Priority (Production Request Editor): 7

PR TYPE	
Weight	.45
<b>On Demand</b>	<b>10</b>
Routine	6
Reprocessing	4
Default	5

PGE	
Weight	.20
MODIS 01	5
<b>MOD09:L2G</b>	<b>3</b>
MOD09:L3	6
MODIS 02	5
Default	5

USER TYPE	
Weight	.15
Operator	5
DAAC Manager	7
Researcher	10
<b>Scientist</b>	<b>8</b>
Default	5

Production Request Editor	
Weight	.20

*The operator can also change the weights given to each list according to the DAACs current production priorities*

$$\begin{aligned}
 \text{Priority} &= 6 * .45 + \\
 & 3 * .20 + \\
 & 8 * .15 + \\
 & 7 * .20 \\
 & = 5.9 = 6
 \end{aligned}$$

*The operator can change any weight or priority on any strategy list.*

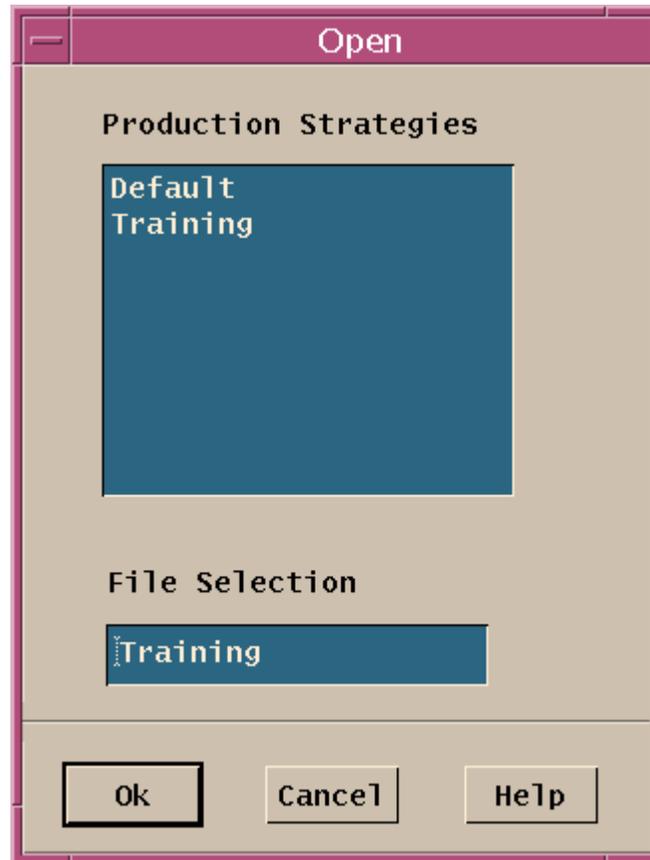
# Defining a Production Strategy



- **Procedure**

- **Select priorities for the values for each of the following three DPR attributes (as needed):**
  - » **PR Type**
  - » **User Type**
  - » **PGE Type**
- **Type weights for the preceding three DPR attributes (as needed)**
- **Type a weight in the Production Request Editor field**
- **Click on the Normalize button**
- **Type delta priority for Late Start Delta (if needed)**
- **Save the Production Strategy**

# Production Strategies “Open” Window



# Production Strategies “Save As” Window



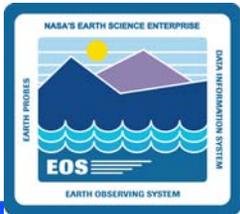
# Reviewing the Current Active Strategy



- **Procedure**

- **Select Options → activeStrategy from the pull-down menu of the Production Strategies GUI**
- **Review the data displayed in the Active Production Strategy window**
  - » **PR Type values/weight**
  - » **User Type data values/weight**
  - » **PGE Type values/weight**
  - » **Production Request Editor weight**
  - » **Late Start Delta Priority value**

# Active Production Strategy Window



Active ProductionStrategy

Current Active Strategy

PR Type	User Type	PGE Type												
Weight: 50.000000	Weight: 0.000000	Weight: 0.000000												
Default: 5	Default: 5	Default: 5												
<table border="1"><thead><tr><th>Value</th><th>Priority</th></tr></thead><tbody><tr><td>Routine</td><td>5</td></tr><tr><td>OnDemand</td><td>7</td></tr><tr><td>Reprocessing</td><td>3</td></tr></tbody></table>	Value	Priority	Routine	5	OnDemand	7	Reprocessing	3	<table border="1"><thead><tr><th>Value</th><th>Priority</th></tr></thead><tbody></tbody></table>	Value	Priority	<table border="1"><thead><tr><th>Value</th><th>Priority</th></tr></thead><tbody></tbody></table>	Value	Priority
Value	Priority													
Routine	5													
OnDemand	7													
Reprocessing	3													
Value	Priority													
Value	Priority													
Find: <input type="text"/>	Find: <input type="text"/>	Find: <input type="text"/>												

Production Request Editor Weight: 50.000000      Late Start Delta Priority: 0

Cancel

# Deleting a Production Strategy



- **Procedure**

- On the Production Strategies GUI click on the option button associated with the Production Strategies field, then highlight (in the option menu) the name of the production strategy to be deleted
  - » Alternatively, it is possible to select File → Open from the pull-down menu, select the desired production strategy from the list on the Open window, and click on the Ok button to open the production strategy
- Select Edit → Delete from the pull-down menu
- Click on the OK button

# Production Strategy Deletion Confirmation Dialogue Box



# Creating a New Production Plan



- **Production Planner creates a plan for production data processing at the DAAC**
  - **selects specific PRs whose DPRs are to be run**
  - **selects PRs from two lists of PRs**
    - » **list of available “Unscheduled” PRs**
    - » **list of “Scheduled” PRs**
  - **uses arrow buttons to move PRs between lists until the “Scheduled” list contains the desired set of PRs that define the new plan**

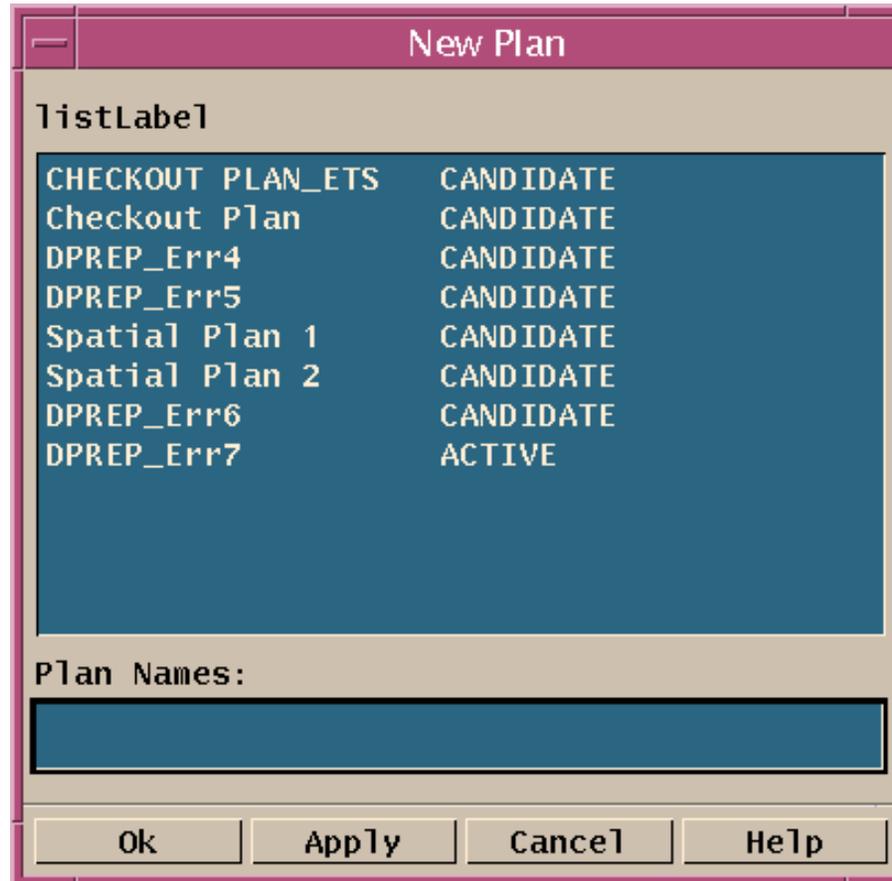
# Creating a New Production Plan (Cont.)



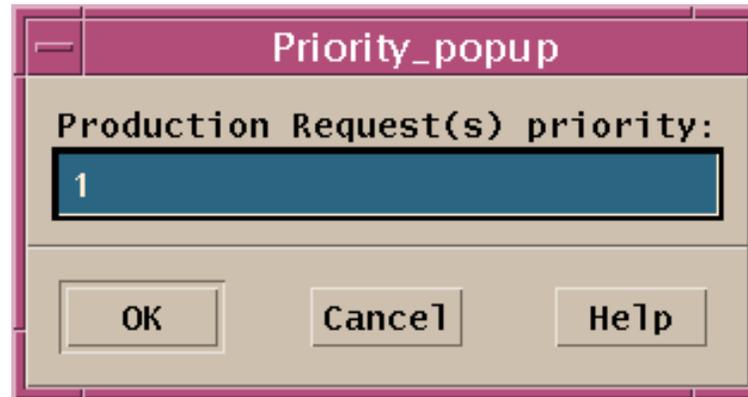
- **Procedure**

- **Select applicable Production Strategy**
- **Move PRs between the Unscheduled and Scheduled lists**
- **Save the plan**
- **Activate the plan if applicable**
- **Save the plan as a baseline plan if applicable**
- **To quit the Planning Workbench GUI: File → Exit**
- **After quitting the Planning Workbench GUI, type the command to shut down the Message Handler, System Name Server, and Resource Model**
- **Verify that the Message Handler, System Name Server, and Resource Model have shut down**

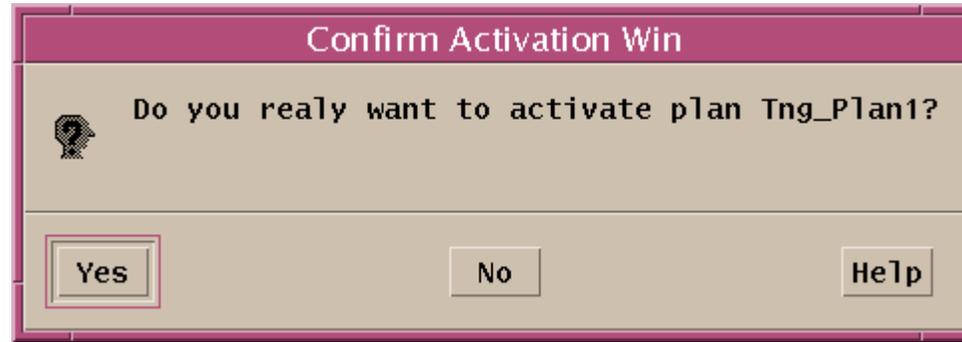
# Planning Workbench “New Plan” GUI



# Planning Workbench Priority Popup Window



# Planning Workbench Confirm Activation Dialogue Box



# Creating a New Production Plan (Cont.)



- **Reactivation/Replanning**
  - Assumes that there is a current plan
  - Current plan may or may not have DPRs associated with it
    - » If there are no DPRs in a plan, it may be because all the DPRs that were in the plan have run to completion
    - » Alternatively, there may never have been any DPRs associated with the plan because an empty plan was activated to cancel DPRs from a previously activated plan
  - Occurs in the following three types of situations:
    - » No DPR in old plan but DPR(s) in new plan
    - » DPR(s) in old plan but not in new plan
    - » DPR(s) in both old plan and new plan

# Creating a New Production Plan (Cont.)



- **Reactivation/Replanning (Cont.)**
  - **No DPR in Old Plan but DPR(s) in New Plan**
    - » **Enter a new plan name (the plan is created using the Planning Workbench GUI)**
    - » **Select the PRs to be included**
    - » **Schedule the PRs to be activated**
    - » **Save the new plan**
    - » **Activate the new plan (new plan is activated and the new DPRs are ready to run)**
    - » **Check on the state of the jobs in AutoSys (new DPRs should be in AutoSys and should begin to run)**

# Creating a New Production Plan (Cont.)



- **Reactivation/Replanning (Cont.)**
  - **DPR(s) in Old Plan but Not in New Plan**
    - » **Verify that DPRs in the active plan are in the Job Management queue (current DPRs should be in the queue)**
    - » **Create and save a new plan without PRs/DPRs to replace the old plan (with DPRs) - do not schedule any PRs to be activated**
    - » **Activate the new plan**
    - » **Verify the state of the new plan's DPRs (use the AutoSys JobScape GUI)**
    - » **Verify the state of the old (existing) plan's DPRs (only jobs that are in the Job Management queue are cancelled; jobs that have already been released to AutoSys are not deleted during the replan - they continue processing to termination)**

# Creating a New Production Plan (Cont.)



- **Reactivation/Replanning (Cont.)**
  - **DPR(s) in Both Old Plan and New Plan**
    - » **Verify that DPRs in the active plan are in the queue (current DPRs should be in the Job Management queue)**
    - » **Enter a new plan name**
    - » **Select the PRs to be included (include relevant PRs used in the old plan; i.e., those PRs with DPRs to be re-prioritized that are in the Job Management queue)**
    - » **Schedule the PRs to be activated**
    - » **Save the new plan**
    - » **Activate plan**
    - » **Verify the priorities of the plan's DPRs in the Job Management queue (new DPRs must be in the Job Management queue; old DPRs must be cancelled)**

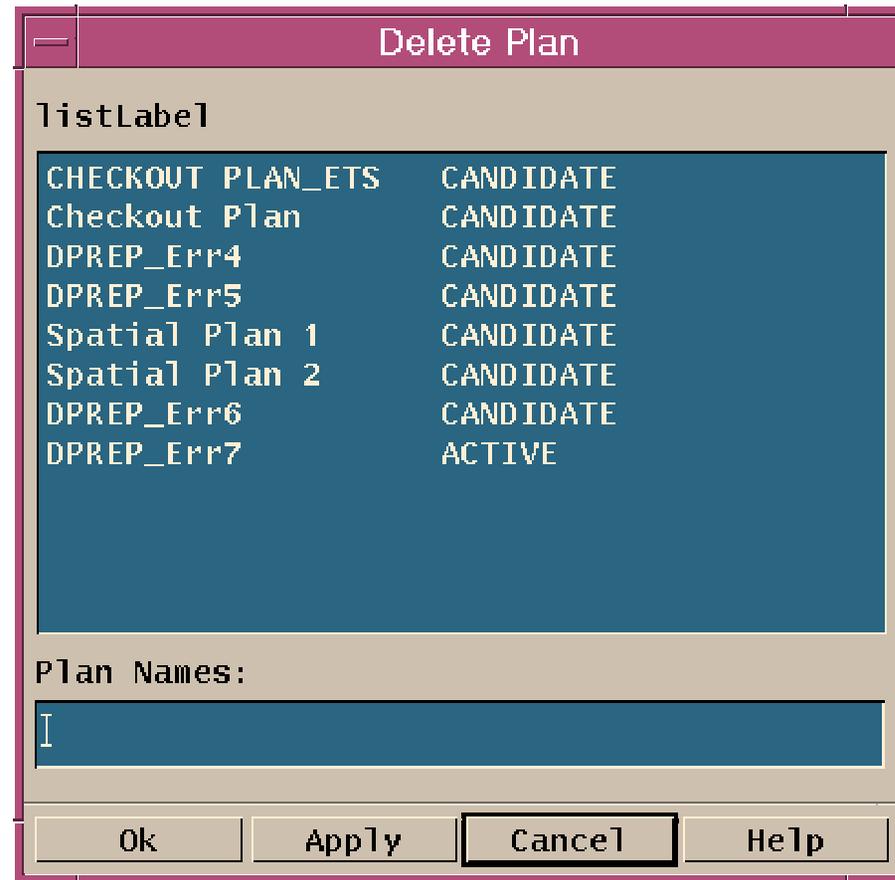
# Deleting a Production Plan



- **Procedure**

- **Select File → Delete on the Planning Workbench GUI**
- **Select (highlight) the production plan to be deleted by clicking on the corresponding name in the list of plans**
- **Click on the OK button**

# Planning Workbench Delete Plan Window



# Reviewing a Plan Timeline



- **Production Plan Timeline**

- **graphic, timeline-oriented depiction of a production plan**
- **displays a set of processing equipment, arranged along the left side of the GUI**
- **displays some period of time across the top edge of the GUI**
- **bars on the timeline represent either...**
  - » **execution of DPRs on processing equipment over a period of time**
  - » **resource reservations for non-production-related purposes (also called “ground events”)**
- **ground events include such activities as testing, preventive maintenance, or system upgrades**

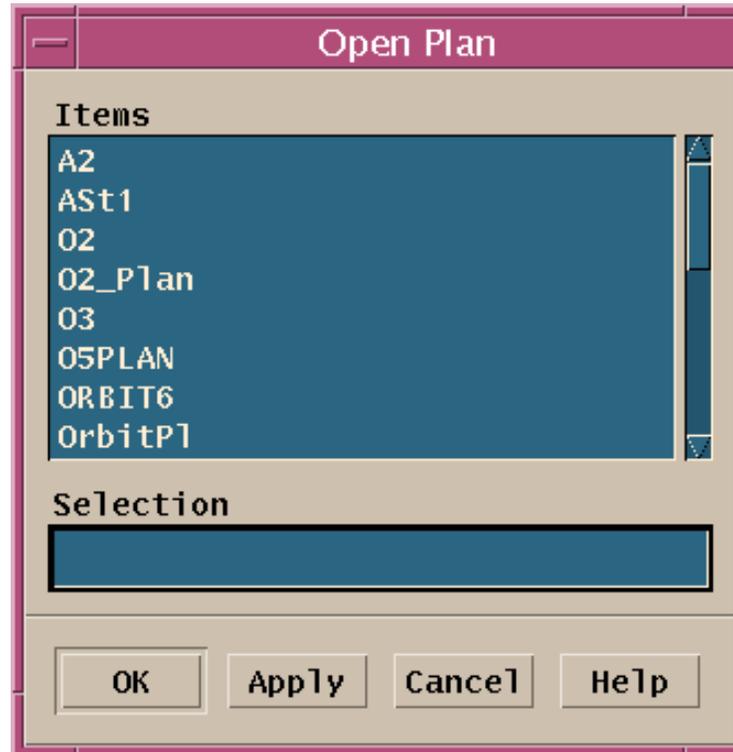
# Reviewing a Plan Timeline (Cont.)



- **Procedure**

- **Adjust the Production Planning Timeline window size and the view of the timeline as necessary**
- **Select a different plan to be viewed if necessary**
- **Adjust the time scale (start and end dates and times) as necessary**
- **Adjust the time span if desired**
- **Adjust the resources to be displayed on the timeline as necessary**
- **Adjust timeline color coding if desired/necessary**

# Planning Timeline Open Plan Window



# Planning Timeline

## Plan Window Edit Window



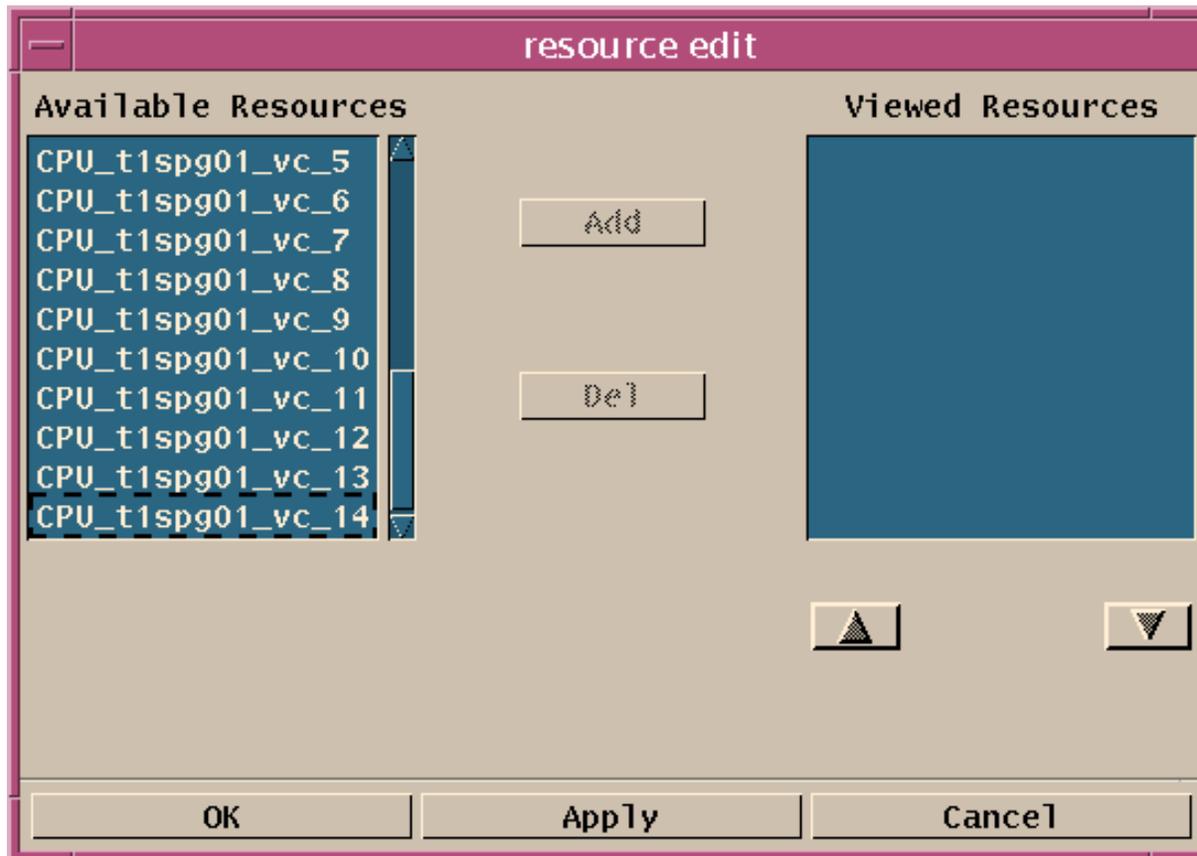
plan window edit

Plan Win Start: 13 FEB 2000 19:57:44

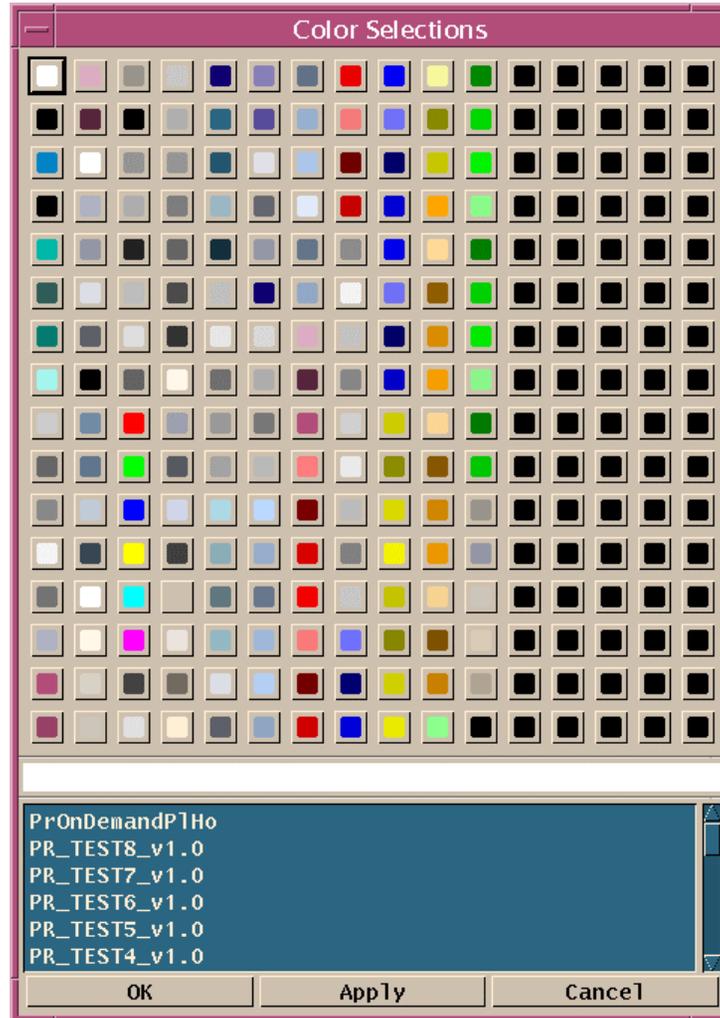
Plan Win End : 13 FEB 2000 19:57:44

OK Apply Cancel

# Planning Timeline: Resource Edit Window



# Planning Timeline Color Selections Window



# Cleaning the PDPS Database and Science Processing Disks



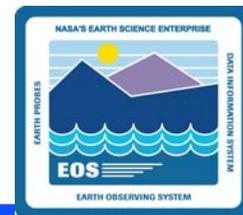
- **At DAACs with a heavy data processing load it is essential to perform PDPS database cleanup and virtual-computer “garbage collection” on a regular basis**
  - **Frequency of cleanup is determined by each DAAC’s needs**
  - **Failure to perform cleanup is likely to lead to performance problems with the Planning and Data Processing Subsystems**

# Cleaning the PDPS Database and Science Processing Disks



- **Recommendation is that the database cleanup and “garbage collection” processes be run in the following order:**
  - Database cleanup
  - “Garbage collection” on the virtual computer(s)
  - Database cleanup (again)
- **If not already done, it is possible to create a script to run the preceding processes**
  - If necessary, processes can be run individually
- **To the extent possible cleanup scripts should be run when the system is relatively quiet:**
  - No PRs/DPRs being created
  - No jobs running in AutoSys

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Cleaning the PDPS Database (EcPIDbClean script)**
  - In the `/usr/ecs/MODE/CUSTOM/utilities` directory on the **Planning/Management Workstation**
  - Cleans up some tables in a PDPS database
  - Tries to delete applicable records in the following order:
    - » **Data Processing Requests based on timeStamp and completionState(SUCC\_DEL)**
    - » **Production Requests that have no associated DPRs**
    - » **Dynamic data granules that are not used by any DPR or by the Data Processing Subsystem**
    - » **PGEs that are marked with a deleteFlag**
    - » **Science Software that has no associated PGE**
  - **Compiles a list of data granules that are not deleted because the Data Processing Subsystem needs to use them**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Cleaning the PDPS Database: Procedure**
  - **Access a terminal window logged in to the Planning/Management Workstation host**
  - **Change directory to the subdirectory (e.g., utilities) containing the production planning startup scripts**
  - **Start the EcPIDbClean script using the following arguments:**
    - » ***MODE***
    - » ***dbuser*** (user name for logging in to interactive structured query language (isql))
    - » ***dbpassword*** (password for isql login)
    - » ***dbserver*** (name of the PDPS database server)
    - » ***months*** (number specifying the removal of records that are older than that number of months)
    - » ***days*** (optional) (number that specifies the removal of records that are older than that number of days)

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Performing Garbage Collection**
  - **EcDpPrGarbageCollectorStart on the Queuing Server**
    - » **Run to delete unneeded files from the science processing disks and update the PDPS database accordingly**
    - » **Actual executable invoked by the script is EcDpPrDeletionClient**
  - **EcDpPrGarbageCollectorStart differs from EcDpPrDeletionClientStart in the following ways:**
    - » **Does not open a separate xterm window**
    - » **Requires specification of a MACHINE\_TO\_COLLECT variable**
    - » **Includes retry logic in case of database deadlock**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Performing Garbage Collection (Cont.): Procedure**
  - Access a terminal window logged in to the Queuing Server host
  - Change directory to the subdirectory (e.g., utilities) containing the production processing startup scripts
  - Start the `EcDpPrGarbageCollectorStart` script using the following arguments:
    - » *MODE*
    - » *machine* - virtual computer (e.g., `x0spg01_vc`) for which garbage collection is being requested
    - » *retries* - number of retries in case of database deadlock
    - » *interval* - amount of time (in minutes) between retries
  - Observe the results as the script runs

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Running the Deletion Server Client**
  - **EcDpPrDeletionClientStart** script on the **Queuing Server**
    - » **Run to delete unneeded files from the science processing disks and update the PDPS database accordingly**
    - » **Executable invoked by the script is EcDpPrDeletionClient**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Running the Deletion Server Client (Cont.): Procedure**
  - Access a terminal window logged in to the Queuing Server host
  - Change directory to the subdirectory (e.g., utilities) containing the production processing startup/utility scripts
  - Start the `EcDpPrDeletionClientStart` script using the following argument:
    - » *MODE*
  - Observe the results as the script runs

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Resolving PDPS Database and Science Processing Disk Content Discrepancies**
  - **EcDpPrRmFilesWOGranules.pl script on the Queuing Server**
    - » **Is run to ensure consistency between the file references in the PDPS database and the files actually staged on the science processing disks**
    - » **Generates a list of files with consistent references among tables in the PDPS database**
    - » **Checks for files on the disk that are not included in the list of files referenced in the PDPS database and either lists the inconsistent files or generates a script to delete them (as requested)**
    - » **Checks to determine whether the disk partitions referenced in the PDPS database actually exist on the disk(s)**
    - » **Removes all file references in the PDPS database that are not included in the list of files**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Resolving PDPS Database and Science Processing Disk Content Discrepancies (Cont.)**
  - **The EcDpPrRmFilesWOGranules.pl script should be run when the system is relatively quiet**
    - » **When no jobs are running in AutoSys**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Resolving PDPS Database and Science Processing Disk Content Discrepancies: Procedure**
  - **Access a terminal window logged in to the Queuing Server host**
  - **Change directory to the subdirectory (e.g., utilities) containing the production processing startup/utility scripts**
  - **Start the EcDpPrRmFilesWOGranules.pl script using the following arguments:**
    - » *dbuser*
    - » *dbpassword*
    - » *MODE*
    - » *dbserver*
    - » *fix [or nofix]*
  - **Observe the results as the script runs**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Saving and/or Resetting the PDPS Database**
  - **Scripts used in saving and resetting the database**
    - » **EcPIDbReset**
    - » **EcPIDbList**
    - » **EcPIDbSave**
  - **Saving the database (using EcPIDbSave) produces one ASCII file (with a “.dat” extension) for each database table**
  - **Resetting the database involves clearing (“wiping out”) the data in the database tables and loading values from a specified “saved database” file**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Saving and/or Resetting the PDPS Database (Cont.)**
  - It is important to take into consideration the consequences of resetting the database before performing the procedure
    - » Removes and replaces all Resource Definitions, Resource Reservations (Ground Events), Production Requests, Data Processing Requests, and Production Plans
    - » Coordinate with all affected parties, including the Resource Planner, Production Planner, and Production Monitors concerning the effects of resetting the database as well as its after-effects (e.g., recreating resource definitions, resource reservations, and production requests)

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Saving and/or Resetting the PDPS Database (Cont.)**
  - **Whenever the PDPS database is reset (not including database cleanup using the EcPIDbClean script) it is also necessary to remove all PLS subscriptions in the Communications Subsystem (CSS) Subscription Server database**
    - » **Production personnel can remove the subscriptions using the Subscription Server GUI (EcSbSubServerGUI) if they have access to the GUI**
    - » **Otherwise, they can request User Services personnel to remove the subscriptions**
    - » **As a result of removing the subscriptions, no subscription notification will come through for existing jobs in the newly loaded database**
    - » **Only new jobs generated using the Production Request Editor will work normally with regard to subscriptions**

# Cleaning the PDPS Database and Science Processing Disks (Cont.)



- **Saving and/or Resetting the PDPS Database: Procedure**

**NOTE: It is important to log in as a user who has “write” permission in the saved\_dumps directory; otherwise it will not be possible to save database contents**

- **Access a terminal window logged in to the Planning/Management Workstation host**
- **Set the ECS\_HOME environmental variable if necessary**
- **Change directory to the subdirectory (e.g., utilities) containing the production planning startup scripts**
- **If desired, save the database**
- **If desired, obtain a listing of saved databases**
- **If desired, reset the database**