

---

# RESOURCE PLANNING

---

**EMD Training**

# Overview of Lesson



- **Introduction**
- **Resource Planning Concepts**
- **Logging in to ECS Hosts**
- **Launching and Shutting Down Resource Planning Applications**
- **Defining Resources**
- **Creating a Resource Reservation Request**
- **Editing a Resource Reservation Request**
- **Reviewing Resource Timelines**
- **Tuning System Parameters**
- **Troubleshooting Resource Planning Problems**

# Overview of Lesson (Cont.)



- **Practical Exercise**
  - **Logging in to ECS Hosts**
  - **Launching Resource Planning Applications**
  - **Shutting Down Resource Planning Applications**
  - **Determining Actual Processing Resources to be Added to the Resource Planning List**
  - **Adding Resources to the Resource Planning List**
  - **Modifying Resources on the Resource Planning List**
  - **Deleting Resources from the Resource Planning List**
  - **Creating a Resource Reservation Request**
  - **Editing/Modifying a Resource Reservation Request**

# Overview of Lesson (Cont.)



- **Practical Exercise (Cont.)**
  - **Validating or Rejecting a Resource Reservation Request**
  - **Approving Resource Reservation Requests**
  - **Committing Resource Reservation Requests**
  - **Deleting a Resource Reservation Request**
  - **Reviewing a Resource Timeline**
  - **Troubleshooting Resource Planning Problems**

# Objectives



- **OVERALL:**
  - Develop proficiency in the procedures that apply to resource planning operations
- **SPECIFIC:**
  - Describe the general steps in the resource planning process
  - Perform the steps involved in...
    - » logging in to ECS hosts
    - » launching resource planning applications
    - » shutting down resource planning applications
    - » determining actual processing resources to be added to the resource planning list
    - » adding resources to the resource planning list
    - » modifying resources on the resource planning list

# Objectives (Cont.)



- **SPECIFIC (Cont.):**
  - **Perform the steps involved in...**
    - » **deleting resources from the resource planning list**
    - » **creating a resource reservation request**
    - » **editing/modifying a resource reservation request**
    - » **validating or rejecting a resource reservation request**
    - » **approving resource reservation requests**
    - » **committing resource reservation requests**
    - » **deleting a resource reservation request**
    - » **reviewing a resource timeline**
    - » **troubleshooting resource planning problems**

# Objectives (Cont.)



- **STANDARDS:**
  - Lesson content (e.g., procedures in the lesson)
  - Mission Operation Procedures for the EMD Project - 611-EMD-001

# Resource Planning Concepts



- **ECS Context**

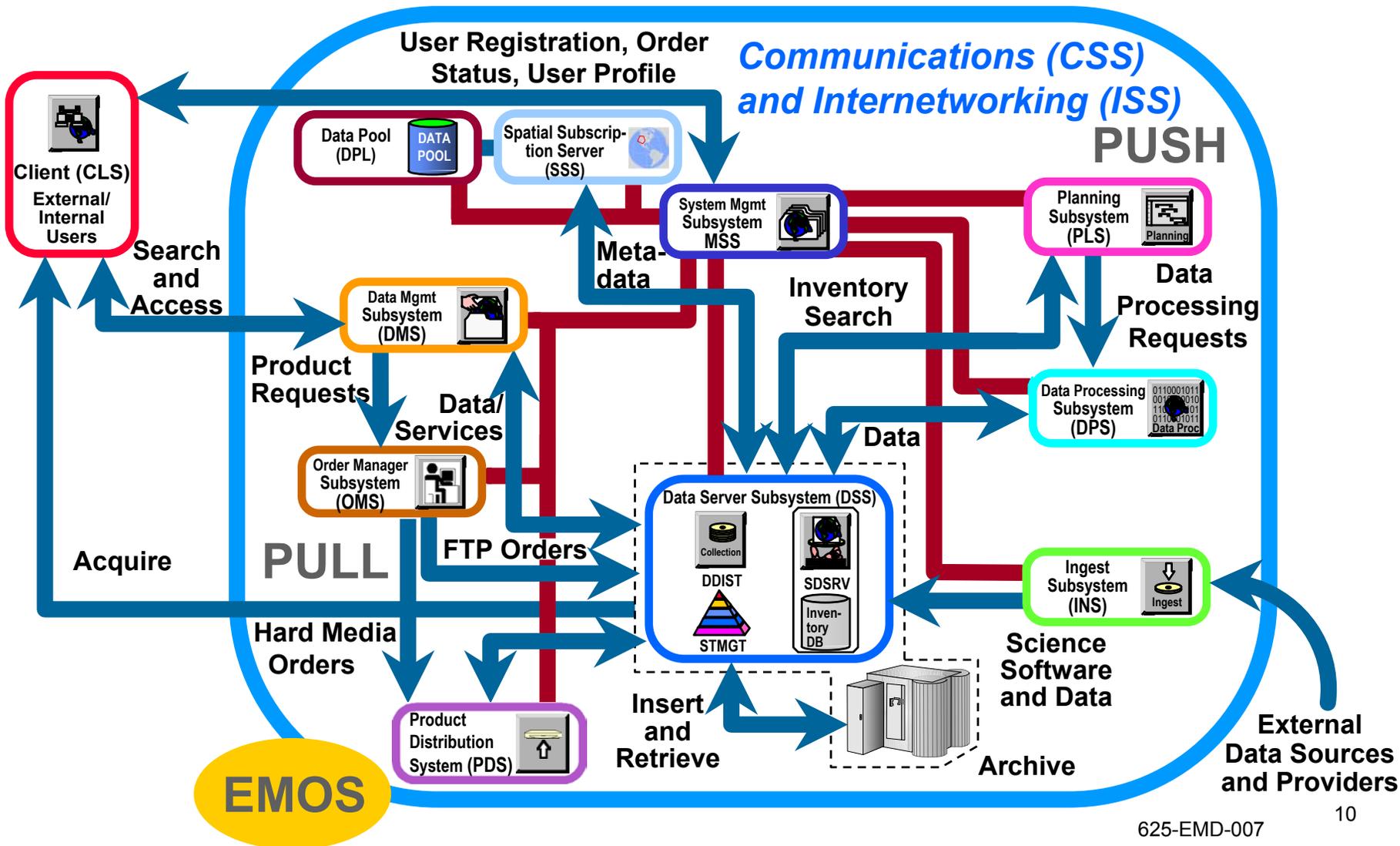
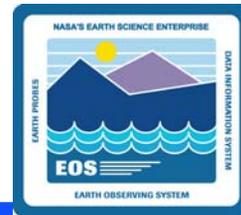
- **ECS resource planning process is accomplished at the Distributed Active Archive Centers (DAACs)**
- **People involved in resource planning activities are...**
  - » **Resource Planner**
  - » **Resource Manager**
  - » **personnel requesting the use of DAAC production resources for non-production-related purposes**

# Resource Planning Concepts (Cont.)



- **ECS Context (Cont.)**
  - **Resource Planner**
    - » defines resources in the Planning and Data Processing Subsystems' (PDPS) database
    - » develops proposed resource plans based on resource reservation requests for non-production-related activities
  - **Resource Manager**
    - » puts a resource plan into effect
  - **Personnel who have a need for Planning Subsystem or Data Processing Subsystem resources**
    - » submit requests for time on specified resources to accomplish the non-routine activities that they plan to undertake

# ECS Context Diagram



**EMOS**

# Resource Planning Concepts (Cont.)

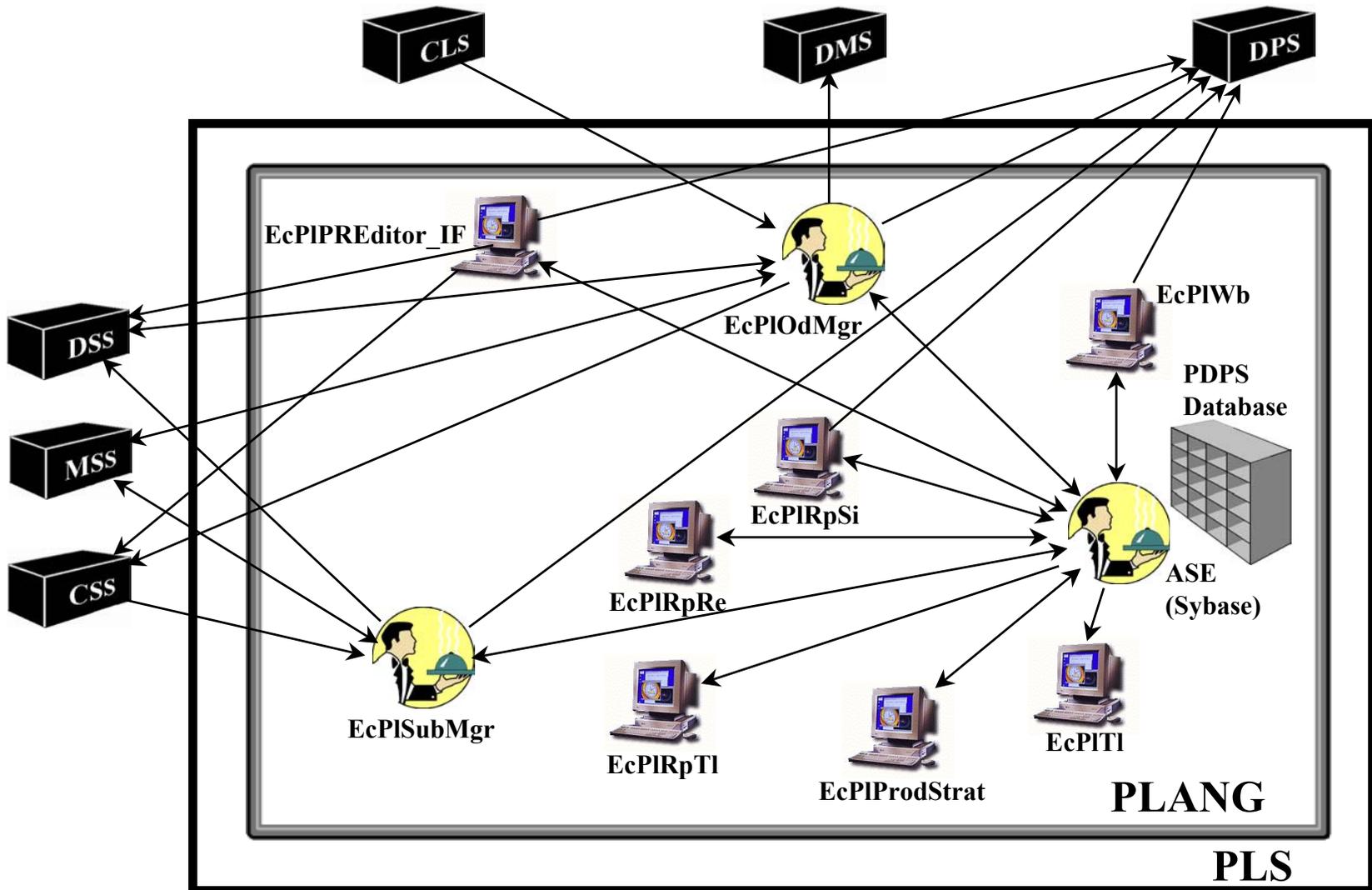


- **Planning Subsystem**

- **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**

# Planning Architecture

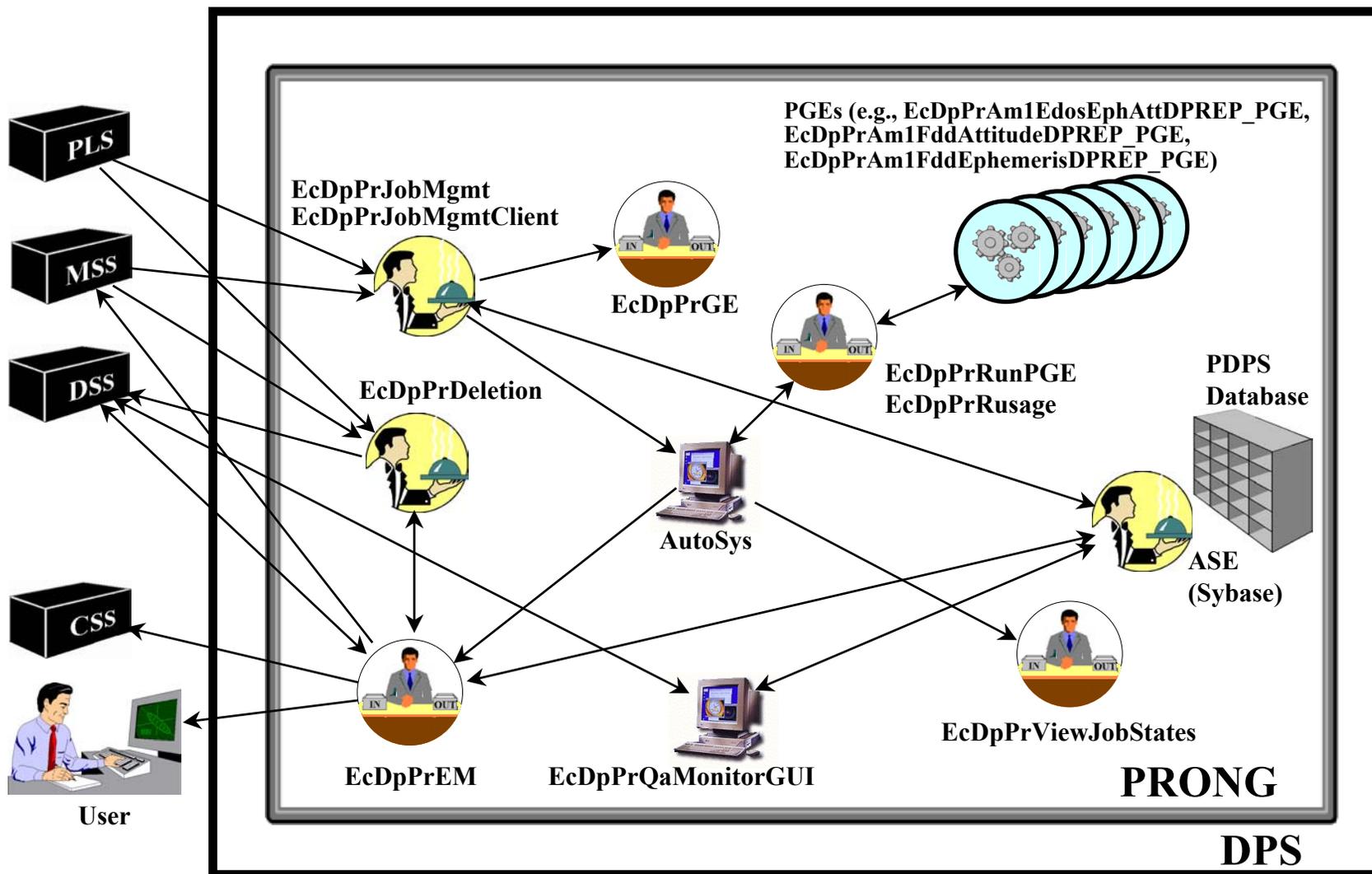


# Resource Planning Concepts (Cont.)



- **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 sets

# Data Processing Architecture (PRONG)



# Resource Planning Concepts (Cont.)



- **Planning Subsystem**
  - **DAAC personnel have access to the resource planning functions of the Planning Subsystem primarily through components of the Resource Planning Workbench in the Planning Subsystem**
    - » **Resource Scheduler (Scheduling Interface)**
    - » **Resource Editor**

# Resource Planning Concepts (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)**

# Resource Planning Concepts (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)**

# Resource 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**

# Resource 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**

# Resource 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**

# Resource 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**

# Resource Planning Concepts (Cont.)



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

# Resource Planning Concepts (Cont.)



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

# Resource Planning Concepts (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

# Resource Planning Concepts (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)**

# Resource Planning Concepts (Cont.)



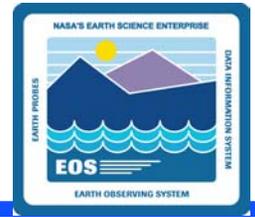
- **PRONG CSCI (Cont.)**
  - **View Job States (EcDpPrViewJobStates)**
  - **Terra Data Preprocessing (DPREP)**
    - » **EcDpPrAm1EdosEphAttDPREP\_PGE**
    - » **EcDpPrAm1FddAttitudeDPREP\_PGE**
    - » **EcDpPrAm1FddEphemerisDPREP\_PGE**
    - » **EcDpPrDumpAttitudeDPREP**
    - » **EcDpPrDumpEphemerisDPREP**

# Resource Planning Concepts (Cont.)



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

# Resource Planning Concepts (Cont.)



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

# Resource 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**

# Resource 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**

# Resource Planning Concepts (Cont.)



- **Resource Definition and Resource Scheduling Processes**
  - **Objective is to define and control reservations for non-routine “ground events”**
    - » **Testing**
    - » **Corrective maintenance**
    - » **Preventive maintenance**
    - » **System upgrades**

# Resource Planning Concepts (Cont.)



- **Resource Definition and Resource Scheduling Processes (Cont.)**
  - **Resource planning affects resources that are scheduled through production planning**
    - » **Resource planning and production planning are interdependent**
  - **Resource planning occurs on a...**
    - » **Biweekly basis for 30-day plans**
    - » **Weekly basis for 10-day plans**
    - » **Daily basis**
  - **Ground events can be entered at any time**
  - **Important point:**
    - » **It is necessary to be aware of the anticipated processing load and upcoming maintenance events about the next month**

# Resource Planning Concepts (Cont.)



- **Resource Definition Process**
  - **ECS resource definitions**
    - » **“Disks”**
    - » **“Virtual computers” (sets of central processing units (CPUs) and associated memory and disks)**
    - » **“Strings” (sets of virtual computers)**
    - » **“Real computers” (hosts that are composed of one or more virtual computers)**
    - » **“AutoSys” (strings associated with the production processing software)**
    - » **Generic “hardware”**

# Resource Planning Concepts (Cont.)



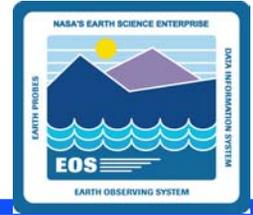
- **Resource Definition Process (Cont.)**
  - **General process used for manually defining production resources**
    - » **Determine what production resources are available**
    - » **Determine the distribution of resources among operating modes**
    - » **Define resources for each mode using the Resource Editor GUI**

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Roles**
  - Resource Planner processes resource reservation requests for ground events
  - Resource Manager commits resource reservations
  - Production Planner sends committed resource reservations (ground events) to Data Processing via the Planning Workbench
  - Production Monitor monitors execution of ground events in processing

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Process**
  - **Personnel who have a need for resources submit requests for time on specified resources to accomplish the non-routine activities that they plan to undertake**
    - » **Depending on DAAC policy, many personnel may have access to the resource planning applications for creating resource reservation requests**
    - » **Alternatively, personnel may have to contact the Resource Planner to have resource reservation requests entered for them**

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Process (Cont.)**
  - **Resource Planner reviews requests for resource reservations to determine if the requests are valid**
    - » **Requests include the activity description, resource(s) required, time period(s) for using the requested resource(s), and comments (e.g., explanation of variance from normal use)**
    - » **Resource Planner may decide to forward the request to a “sponsor” for validation (sponsor is someone who evaluates a resource reservation request based on relevant expertise)**
  - **If the Resource Planner or sponsor determines that the request to reserve the resource is valid, the Resource Planner “approves” it along with all other requests that have been validated**
    - » **The set of all validated resource reservation requests is considered a draft Resource Plan**

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Process (Cont.)**
  - The scheduling software identifies conflicts (if any) in the draft Resource Plan and alerts the Resource Planner to the problem(s)
  - If possible, the Resource Planner resolves all conflicts before presenting the proposed plan to the Resource Manager to have the resources committed
    - » When resolving conflicts, the Resource Planner may have to consult with resource requesters and the Resource Manager to ensure that the reserved resources will not have adverse effects on the DAAC's high-priority events
  - When the Resource Planner has achieved a conflict-free plan, it is presented to the Resource Manager to be implemented

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Process (Cont.)**
  - The Resource Manager "commits" the resource plan, which signals the Planning Subsystem that the plan can be implemented
    - » Committing a plan actually involves committing all of the individual approved resource reservation requests that collectively make up the plan
  - All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing
    - » Resource reservations/ground events cannot take effect until they have been sent to Data Processing as part of an activated production plan

# Resource Planning Concepts (Cont.)



- **Resource Scheduling Process (Cont.)**
  - In Data Processing a ground event job for each resource reservation is sent to the specified resource(s) at the indicated start time
    - » If a data processing job is already using the specified resource(s) at the ground event's scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job

# 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 Resource Planning 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 Resource Planning Applications



- **Resource Scheduler GUI (Scheduling Interface)**
- **Resource Editor GUI**
- **Message Handler**
- **System Name Server**
- **Resource Model**
- **Resource Reservation Planning Master Timeline GUI**

# Launching Resource Planning Applications (Cont.)



- **Access**
  - **Submitting resource reservation requests**
    - » all ECS personnel who may need to use system resources
  - **Validating resource reservation requests**
    - » sponsors
  - **All other functions**
    - » Resource Planner
    - » Resource Manager
- **Use UNIX command line to gain access to graphical user interfaces (GUIs)**

# Launching Resource Planning Applications (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 resource planning startup scripts**
  - **Type command to start Message Handler, System Name Server and Resource Model**
  - **Type command to start Resource Editor**
  - **Type command to start Resource Scheduler**

# Message Handler GUI



```
Message Handler
[Engineering] [Information] [Warning] [Internal] Filter Clients
feb-01 07:32:28 <EcP1RpRm2-TS2> information (INFO) P1RpRmApp1::loadPools-Loading Reservations from Database
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) P1RpDbReservationCfg::dumpActPool-pool is empty
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) P1RpRmApp1::loadPools-Finished loading Reservations from Database
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) P1RpRmApp1::loadPools-Loading Resources from Database
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) String Resource: t1spg03_string
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) Autosys Resource: VAT
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) Local Disk Resource: t1spg03_disk
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) Computer Resource: t1spg03_vc
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) Hardware Resource: t1spg03
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) P1RpRmApp1::loadPools-Finished loading Resources from Database
feb-01 07:32:30 <EcP1RpRm2-TS2> information (INFO) P1RpRmApp1::loadPools-Generating Plan
```

# Resource Editor



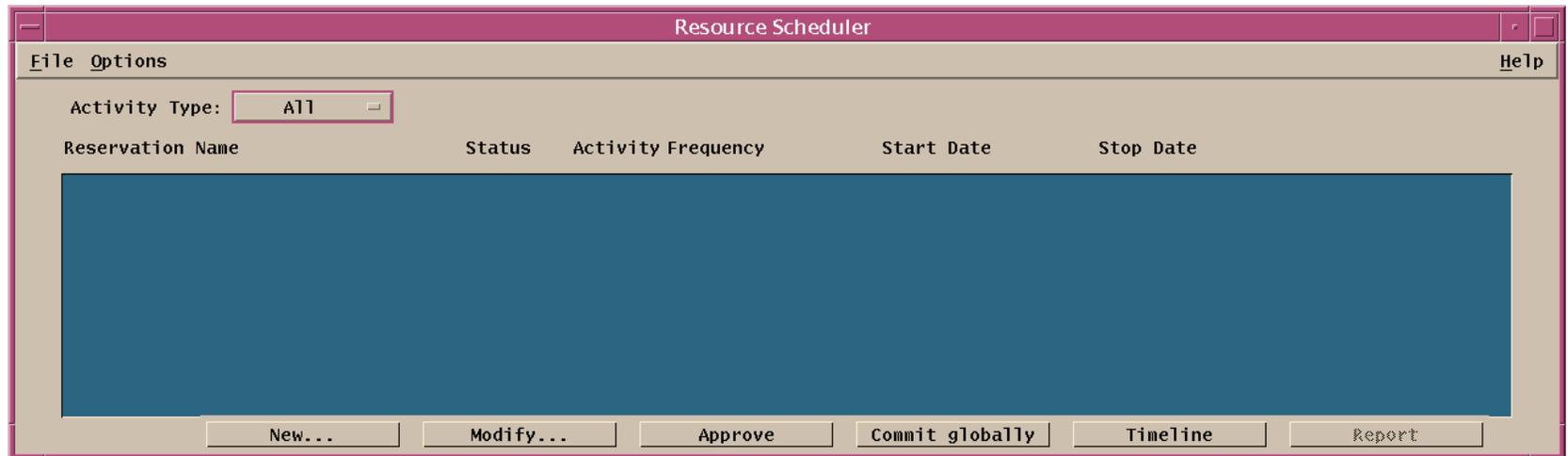
Resource Editor

File Help

Resource Type:

Resource Name	Type	Activity
VAT	AUTOSYS	production
VT2	AUTOSYS	production
t1aqg02	REALCOMP	production
t1aqg02_disk	DEVICE	production
t1aqg02_string	VIRTUAL	production
t1aqg02_vc	MACHINE	production
t1spg01	REALCOMP	production
t1spg01_disk	DEVICE	production
t1spg01_string	VIRTUAL	production
t1spg01_vc	MACHINE	production

# Resource Scheduler



# Shutting Down Resource Planning Applications



- **Shut down the resource planning processes when resource planning activities have been completed**
  - Resource Editor
  - Resource Scheduler
  - Message Handler
  - System Name Server
  - Resource Model
- **Allows other operators to gain access to resource planning applications**

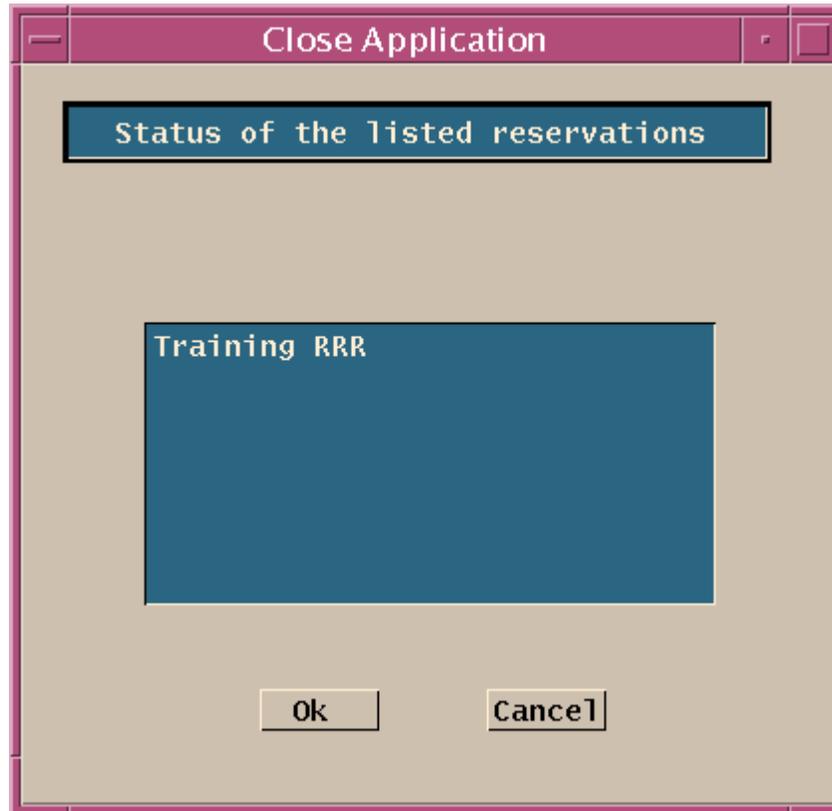
# Shutting Down Resource Planning Applications (Cont.)



- **Procedure**

- **Quit (File → Exit) Resource Editor**
- **Quit (File → Exit) Resource Scheduler**
- **Access UNIX command shell**
- **Type command to shut down resource planning applications**
- **Verify that resource planning applications are no longer running in the applicable mode**
  - » **Terminate processes individually if necessary**

# Close Application Dialogue Box

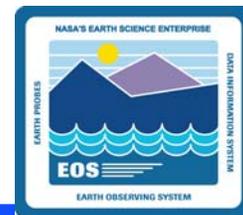


# Defining Resources



- **Resource Editor**
  - **Allows the authorized user to...**
    - » **Add or delete resources**
    - » **Modify the characteristics of resources**
  - **Makes modifications to the resource planning list in the PDPS database**

# Adding or Modifying Resources



- **The Resource Editor allows the authorized operator to define resources**
- **Resource categories:**
  - **Disks**
    - » **Disk partitions that are associated with and provide temporary data storage for the input and output files used in processing**
  - **Virtual Computers**
    - » **Virtual computers composed of CPUs, random-access memory (RAM), and associated-disk(s)**
  - **Real Computers**
    - » **Physical computing devices (hosts), each of which contains one or more CPUs**

# Adding or Modifying Resources (Cont.)



- **Resource categories (Cont.):**
  - **Strings**
    - » **Sets of one or more virtual computers**
  - **AutoSys**
    - » **Identifies the string(s) of virtual computers used by the production processing software**
  - **Hardware**
    - » **Any type of equipment that is not defined as a computer or disk may be defined as “hardware”**

# Adding or Modifying Resources (Cont.)



- **The ECS Operational Readiness Plan for Release 2.0 (603-CD-003)**
  - **Initially disk partitions at the DAACs are to be split among the operating modes**
    - » **OPS – 60%**
    - » **TS1 - 20%**
    - » **TS2 - 20%**
  - **It may be advantageous to reserve some nominal percentage of the disk as a safety buffer**
    - » **e.g., two to five percent**
  - **Critical to ensure that the sum of the disk space assigned to the various modes is no more than the total disk space available**

# Adding or Modifying Resources (Cont.)



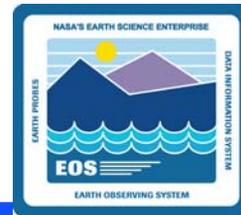
- **CPUs and RAM should be allocated among modes**
  - **No one-to-one mapping of CPU allocation with actual CPUs on the science processor**
  - **Actual CPU usage during processing is limited by the operating system (OS)**
    - » **If ten CPUs have been specified for a particular mode, only ten Data Processing Requests (DPRs) can be running the Execute job at a given time**
    - » **What is really being defined is the maximum number of DPRs that will execute at a given time**
  - **Important to monitor the load on each science processor**
    - » **CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs**

# Adding or Modifying Resources (Cont.)



- **Random-access memory (RAM) is subject to the same considerations as CPUs**
  - **RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor**
  - **The OS takes care of true CPU and RAM allocation**

# Adding or Modifying Resources (Cont.)



- **Determining Actual Processing Resources**
  - **The following types of information are needed:**
    - » **Host names [“real computers”]**
    - » **Number of processors [CPUs] available on each host**
    - » **Operating System (OS) for each host**
    - » **Memory [RAM] on each host**
    - » **Total disk space**
    - » **AutoSys instance(s) at the DAAC**

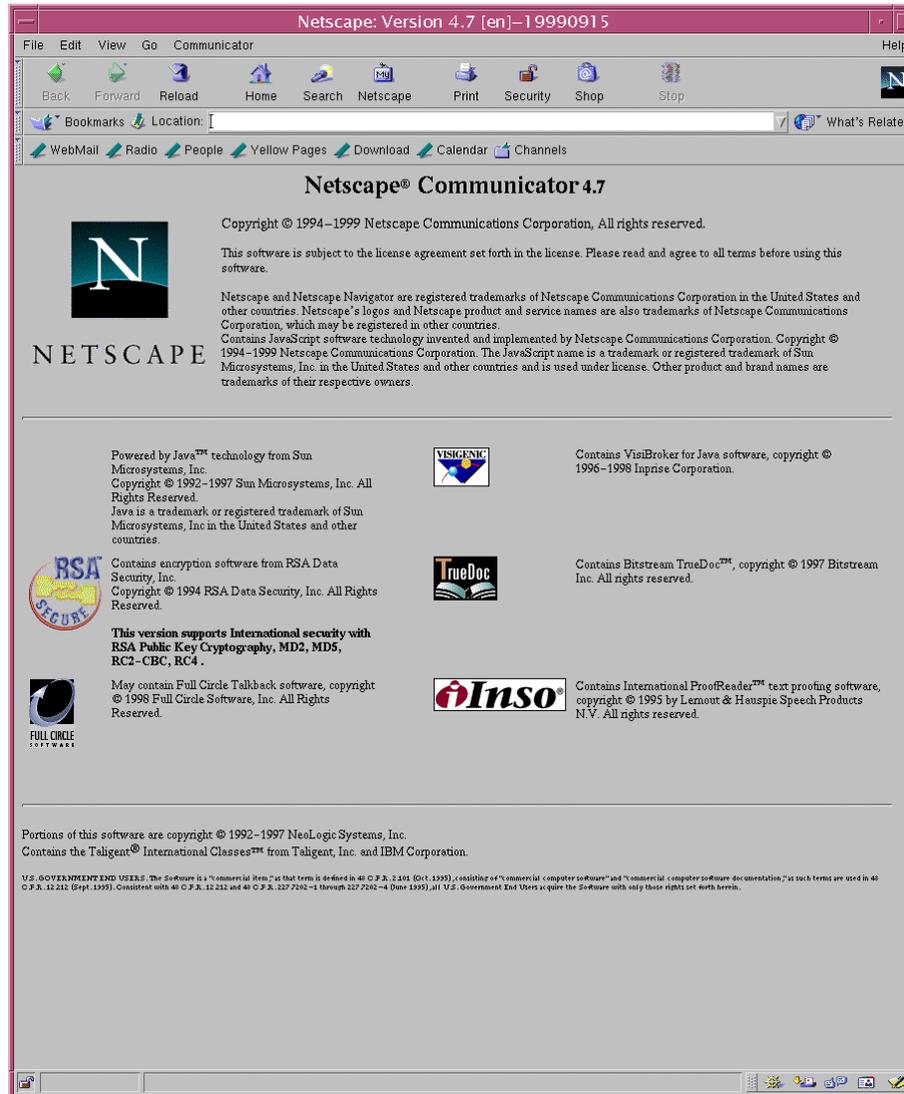
# Adding or Modifying Resources (Cont.)



- **Procedure**

- **Access a terminal window logged in to the applicable Science Processor**
- **Change to the disk mount point (subdirectory)**
- **Identify the disk name and size by changing to the disk mount point and typing `df -k`.**
- **Identify the number of processors (CPUs) and amount of RAM (type `hinv`)**
- **Launch Netscape**
- **Identify the Operating System by selecting the as-built file name corresponding to the desired host at the relevant DAAC (e.g., `x0spg11.asbuilt.html`)**
- **Access a terminal window logged in to the applicable Queuing Server host**
- **Identify the AutoSys instance (in the “autouser” directory)**

# Netscape Web Browser



# ECS Baseline Information System Web Page

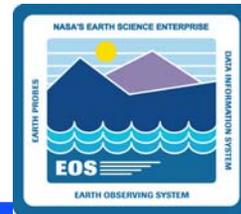


# Adding or Modifying Resources (Cont.)



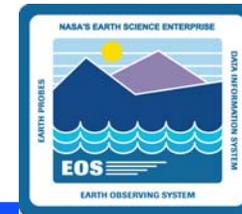
- **Example: Definition of Resources at DAAC X**
  - **Two science processors**
    - » **x0spg01**
    - » **x0spg02**
  - **In both cases disk space is...**
    - » **413,394,688 kilobytes**
    - » **413,394.688 megabytes**
  - **3% of each disk reserved as a safety buffer**
    - » **Each disk has a total of 400,992.847 megabytes functionally available for operational use**

# Adding or Modifying Resources (Cont.)



- **General Resource allocation**
  - OPS mode
    - » split among x0spg01 and x0spg02
  - TS1 mode
    - » all from x0spg01
  - TS2 mode
    - » all from x0spg02
- **CPU allocation**
  - one CPU of each science processor reserved for use by the operating system
  - total number of CPUs in both x0spg01 and x0spg02 is 16 each
- **RAM allocation**
  - total 2048 megabytes

# Adding or Modifying Resources (Cont.)



## Example: Definition of Resources at DAAC X

Resource Type	Resource Name	Activity	Partition Size [mega-bytes]	Block Size [bytes]	CPUs	RAM [mega-bytes]	Oper Sys	Associated Disks / Computers / Strings
<b>OPS Mode</b>								
Disk	x0spg01_disk OPS	Production	240595.708	1024				
Disk	x0spg02_disk OPS	Production	240595.708	1024				
Computer	x0spg01_vc OPS	Production			9	1228	IRIX 6.5.17	x0spg01_disk OPS
Computer	x0spg02_vc OPS	Production			9	1228	IRIX 6.5.17	x0spg02_disk OPS
Real Computer	x0spg01	Production						x0spg01_vc OPS
Real Computer	x0spg02	Production						x0spg02_vc OPS
String	string OPS	Production						x0spg01_vc OPS x0spg02_vc OPS
Autosys	FMR	Production						string OPS
<b>TS1 Mode</b>								
Disk	x0spg01_disk TS1	Production	160397.138	1024				
Computer	x0spg01_vc TS1	Production			6	819	IRIX 6.5.17	x0spg01_disk TS1
Real Computer	x0spg01	Production						x0spg01_vc TS1
String	string TS1	Production						x0spg01_vc TS1
Autosys	FMR	Production						string TS1
<b>TS2 Mode</b>								
Disk	x0spg02_disk TS2	Production	160397.138	1024				
Computer	x0spg02_vc TS2	Production			6	819	IRIX 6.5.17	x0spg02_disk TS2
Real Computer	x0spg02	Production						x0spg02_vc TS2
String	string TS2	Production						x0spg02_vc TS1
Autosys	FMR	Production						string TS2

# Adding a Resource



- **Procedure**

- **Select the appropriate Resource Type from the option button on the Resource Editor**
  - » **Disk**
  - » **Virtual computer**
  - » **Real computer**
  - » **String**
  - » **AutoSys**
  - » **Hardware (generic hardware)**
- **Click on the New... button**
- **Perform the subordinate procedure corresponding to the selected Resource Type**
  - » **Selection of Resource Type determines which GUI appears when the New... button is activated**

# Resource Editor



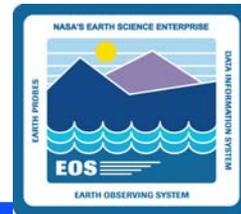
Resource Editor

File Help

Resource Type:

Resource Name	Type	Activity
VAT	AUTOSYS	production
VT2	AUTOSYS	production
t1aqg02	REALCOMP	production
t1aqg02_disk	DEVICE	production
t1aqg02_string	VIRTUAL	production
t1aqg02_vc	MACHINE	production
t1spg01	REALCOMP	production
t1spg01_disk	DEVICE	production
t1spg01_string	VIRTUAL	production
t1spg01_vc	MACHINE	production

# Disk Partition Details GUI



**Disk Partition Details**

Resource Name:

Activity:

Partition Size:  MBytes      Block Size:  Bytes

Comments:



# Virtual Computer Details GUI

Virtual Computer Details

Resource Name:

Activity:

Number of CPUs:

Total Ram:  MBytes

Operating System:

Disks		Associated Disks
<input type="text" value="t1aqq02_disk"/>	<input type="button" value="▶"/>	<input type="text" value="t1spg01_disk"/>
	<input type="button" value="◀"/>	

Comments:

# Real Computer Details GUI



RealComputer Details

Resource Name:

Activity:

Computers

t1aqg02\_vc

Associated Computers

t1spg01\_vc

Comments:

Save Cancel

# String Details GUI



String Details

Resource Name:

Activity:

Computers

Associated Computers

Comments:

# Autosys Details GUI



Autosys Details

Resource Name: VAT

Activity: production

Strings

t1aqq02\_string

Associated Strings

t1spg01\_string

Comments:

Save Cancel

# Hardware Details GUI



Hardware Details

Resource Name:

Activity:

Comments:

# Modifying a Resource



- **Procedure**
  - **Select the resource to be modified from the list displayed on the Resource Editor**
  - **Click on the Modify... button**
  - **Perform the subordinate procedure corresponding to the selected Resource Type**
    - » **Selection of Resource Type determines which GUI appears when the Modify... button is activated**
    - » **Make modifications in the same manner as entries were made when Adding a Resource**

# Defining Resources (Cont.): Procedure



- **Deleting a Resource (Procedure)**
  - **Select the resource to be deleted from the list on the Resource Editor**
  - **Click on the “Delete” button**
  - **Click on the “Ok” button in the confirmation dialogue box**

# Delete Confirmation Dialogue Box



# Creating a Resource Reservation Request



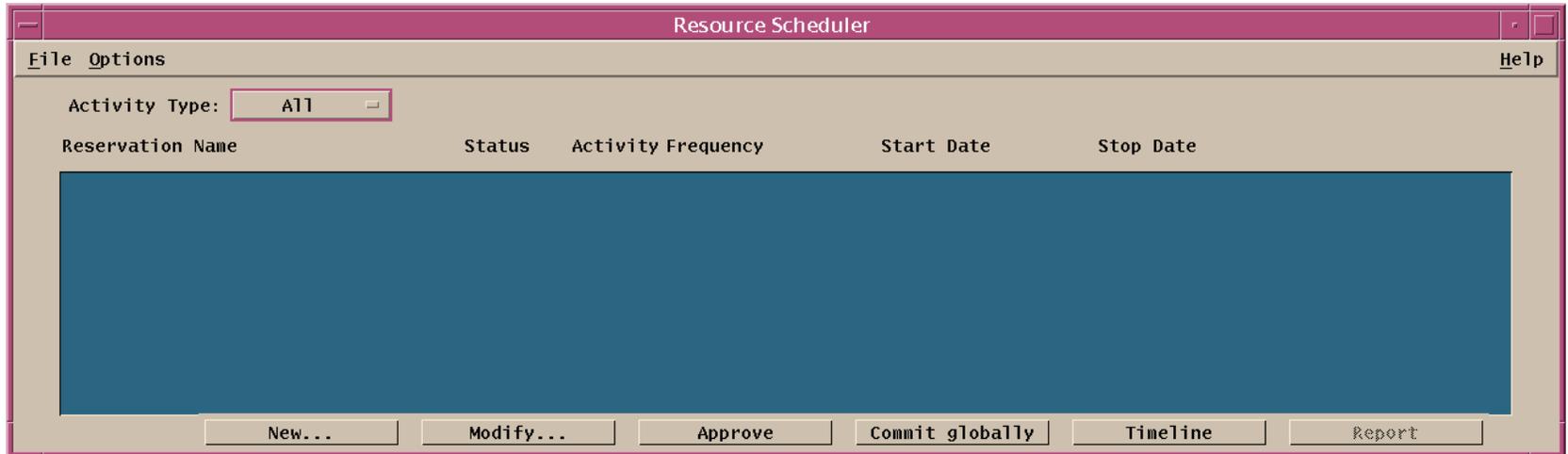
- **Resource Reservation Request describes....**
  - **Activity for which the request is being made**
  - **Resources to be dedicated to the activity**
  - **When/how often the activity will occur**

# Creating a Resource Reservation Request (Cont.)



- **Procedure**
  - **Gain access the Resource Reservation Request Edit/Definition GUI from the Resource Scheduler**
    - » **Click on the New... button**
  - **Specify activity for which the request is being prepared (include a description)**
  - **Set the priority of the requested activity**
  - **Select resources (separate procedure section)**
  - **Enter duration information**
  - **Select frequency (separate procedure section)**
  - **Enter relevant comments**
  - **Save the request**

# Resource Scheduler



# Resource Reservation Request Edit GUI



Resource Reservation Request Edit/Definition – New

Request Name: [ ]

Edited Date: 02/06/2000 At 18:05:23

Originator: [ ]

Sponsor: [ ]

Activity: production [ ] Priority: 0 [ ]

Description: [ ]

[ Resource ... ] [ Interval ... ]

Start Day as "MM/DD/YYYY" 02/06/2000 Start Time as "HH:MM:SS" 18:05:23

Stop Day as "MM/DD/YYYY" 02/06/2000 Stop Time as "HH:MM:SS" 18:05:23

Frequency: Once [ ]

Rejected  Validated Status: new [ ]

Comments: [ ]

[ Save ] [ Clear ] [ Cancel ]

# Resources Selection GUI



Resources Selection

Request Name:

Resources:

- VAT
- VT2
- t1aqq02
- t1aqq02\_string
- t1aqq02\_vc
- t1spg01
- t1spg01\_string
- t1spg01\_vc

Selected Resources:

▶

◀

Ok Cancel

# Selecting Frequency



- **Procedure**

- **Click on the Frequency option button and select the appropriate frequency**
- **If Every\_?\_Days was selected, type the number of days between actions in the field to the right of the Frequency button**
- **Return to the appropriate procedure**
  - » **Creating a Resource Reservation Request**
  - » **Editing a Resource Reservation Request**

# Editing a Resource Reservation Request



- **Editing may be needed in response to any of the following factors (for example):**
  - activities related to evaluation of the resource reservation request for validation purposes
  - change in the activity/event for which the resource reservation request was prepared
  - addition or deletion of resources
  - modification of intervals for recurring ground events
  - resource conflicts

# Editing a Resource Reservation Request (Cont.)



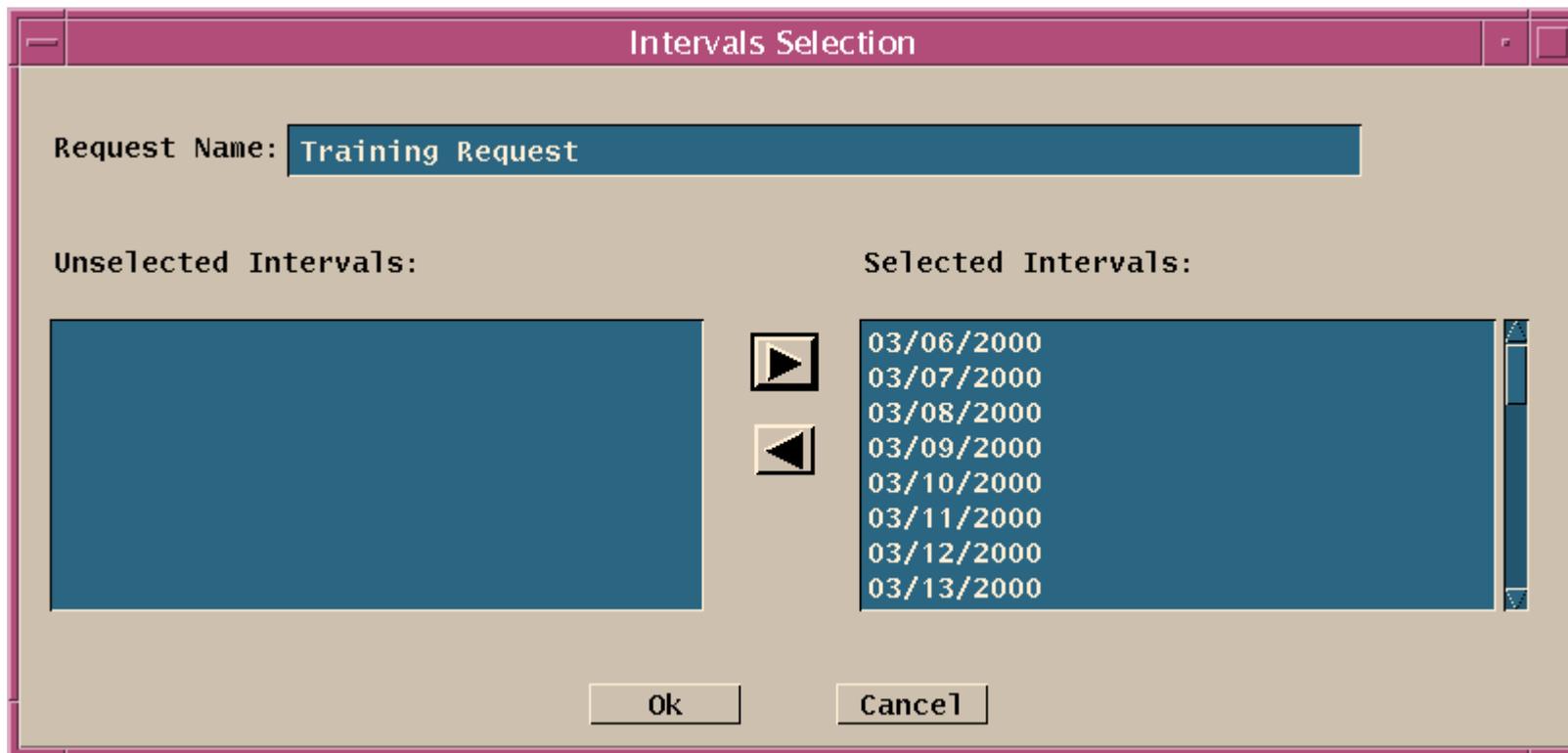
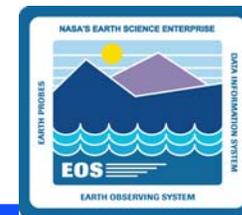
- **Procedure**
  - **select the resource reservation request to be modified from the list on the Resource Scheduler**
  - **gain access the Resource Reservation Request Edit/Definition GUI**
    - » **click on the Modify... button**
  - **make modifications in the same manner as entries were made when Creating a Resource Reservation Request**
    - » **Status will revert to “new” when the edited resource reservation request is saved if certain types of modifications have been made (e.g., changes in the selected resources or start/stop date/time)**
    - » **deselect intervals (separate procedure section) if applicable**

# Deselecting Interval



- **Procedure**
  - **Click on the Interval... button on the Resource Reservation Request Edit/Definition GUI**
  - **Move dates between lists**
    - » **Selected Intervals**
    - » **Unselected Intervals**
  - **Click on the OK button**

# Intervals Selection GUI



# Editing a Resource Reservation Request (Cont.)



- **All resource reservation requests must be validated and approved before scheduling**
- **Validation**
  - **Is the request complete and reasonable?**
  - **Evaluation may be made by a “sponsor”**
  - **“Validated” and “Rejected” buttons on the Resource Reservation Request Edit/Definition GUI**

# Editing a Resource Reservation Request (Cont.)



- **Approval process**
  - **Sponsor has validated request**
  - **Resource Planner submits request to PDPS for approval**
  - **If the system detects conflicts...**
    - » **A dialog box pops up indicating that there are conflicts to be resolved**
    - » **The Resource Planner resolves the conflicts, (in consultation with the requesters and Resource Manager as needed) making modifications to resource reservation requests as necessary**
    - » **System approves a resource reservation request only when there are no scheduling conflicts**

# Editing a Resource Reservation Request (Cont.)



- **Approval process (Cont.)**
  - **Later the Resource Manager will review and “commit” the set of approved resources**
    - » **Committed resource reservations/ground events cannot take effect until they have been sent to Data Processing as part of an activated production plan**

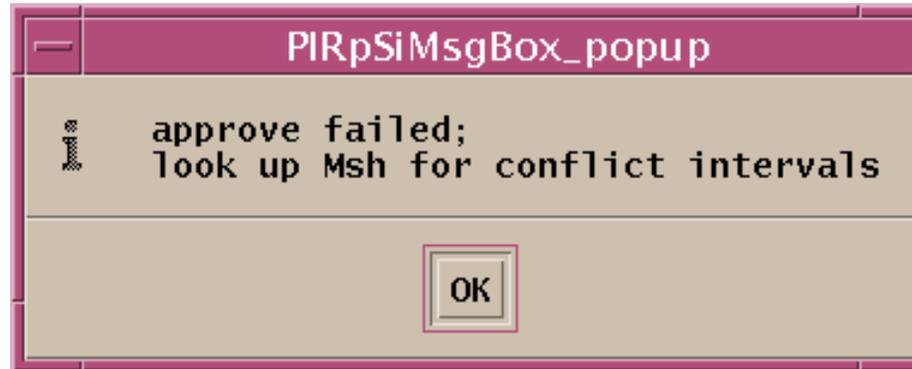
# Approving a Resource Reservation Request



- **Procedure**

- **Select the resource reservation request to be approved from the list on the Resource Scheduler**
- **Click on the “Approve” button on the Resource Scheduler**
  - » **Request status changes to “approved” unless there are conflicts**
- **Resolve conflicts (modify or delete resource reservation requests as necessary)**

# PIRpSiMsgBox\_popup (Approval Failed) Dialogue Box



# Committing and Deleting Resource Reservation Requests



- **Committing Resource Reservation Requests**
  - Validated
  - Approved
  - No conflicts
  - “Commit globally” button on the Resource Scheduler
  - All resource reservation requests with a status of “Approved” have their status changed to “Committed” at the same time
  - All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing
    - » Committed resource reservations/ground events cannot take effect until they have been sent to Data Processing as part of an activated production plan

# Committing and Deleting Resource Reservation Requests



- **Committing Resource Reservation Requests (Cont.)**
  - In Data Processing a “ground event” job for each resource reservation is sent to the specified resource(s) at the indicated start time
    - » If a data processing job is already using the specified resource(s) at the ground event’s scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job

# Committing and Deleting Resource Reservation Requests



- **Deleting a Resource Reservation Request: Procedure**
  - **Select resource reservation request to be deleted**
  - **Select File → Delete**
    - » **Entry for the resource reservation request is deleted from the GUI**

# Reviewing Resource Timelines



- **Reviewing a Resource Timeline**
  - **“Timeline” button on the Resource Scheduler**
    - » **set of resources, arranged along the left side of the screen**
    - » **period of time is indicated across the top edge of the screen**
    - » **use of a resource over a period of time is represented by “resource reservation” bars across the screen**
    - » **bar represents a time period during which a reservation has been made for the resource**
    - » **when there is no reservation affecting a particular resource, it is available for its default activity**

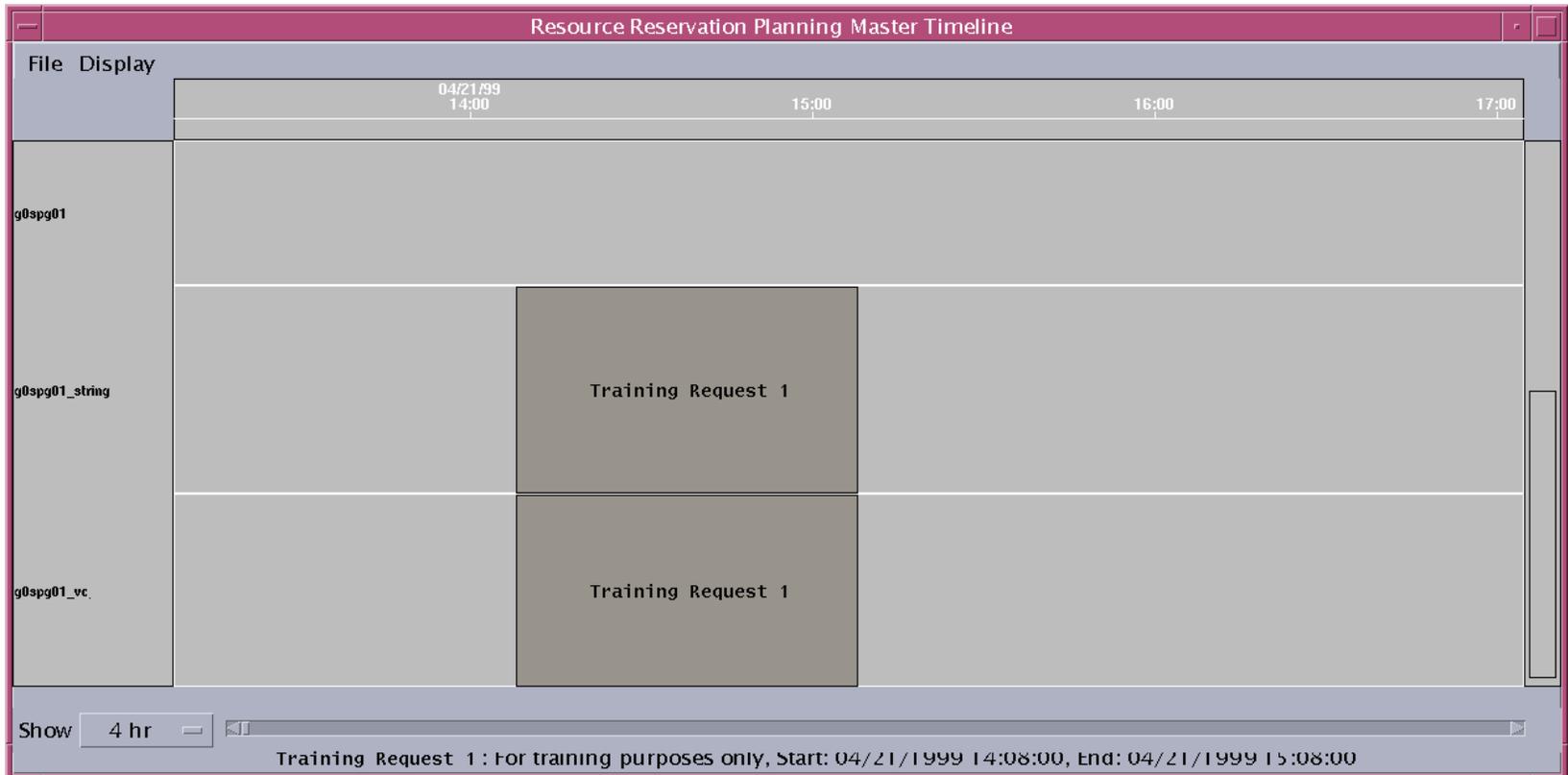
# Reviewing a Resource Timeline



- **Procedure**

- **Click on the “Timeline” button on the Resource Scheduler**
- **Adjust the Resource Timeline window size and view as necessary**
- **Change the time scale if necessary**
- **Change the time span if necessary**
- **Change the set of resources to be displayed if necessary**
- **Change the color coding of the timeline if desired**

# Resource Reservation Planning Master Timeline GUI



# Resource Planning Timeline: Plan Window Edit Window



plan window edit

Plan Win Start: 21 APR 1999 13:08:00

Plan Win End : 21 MAY 1999 15:08:00

OK Apply Cancel

# Resource Planning Timeline: Resource Edit Window



# Resource Planning Timeline: Color Selections Window



# Tuning System Configuration Parameters



- **System parameters may be subject to control by Configuration Management (CM)**
  - When requesting a change to system parameters, the CM process at the particular site must be followed (if applicable)
- **Two databases where parameters can be set:**
  - PDPS database
  - Configuration Registry database

# Tuning System Configuration Parameters (Cont.)



- **Configuration Registry**
  - Provides a single interface (via a Sybase server) for retrieving configuration attribute-value pairs for ECS servers from the Configuration Registry Database
  - When ECS servers are started they access the Configuration Registry Database to obtain needed configuration parameters
  - Database Administrator has access to a Configuration Registry GUI for viewing and editing configuration data in the database
  - It is necessary to coordinate with the Database Administrator when changes to configuration parameters are needed
  - Changes to configuration-controlled parameters are subject to approval through the site CM process

# Tuning System Configuration Parameters (Cont.)



- **Default and adjusted values assigned to system parameters vary from site to site**
  - For guidance concerning the assignment of values to parameters included in the Configuration Registry refer to document 910-TDA-022, Custom Code Configuration Parameters for ECS
    - » Document is available at <http://cmdm.east.hitc.com/baseline/> under “Technical Documents”

# Tuning System Configuration Parameters (Cont.)



- **Parameters whose values may be modified to enhance system functioning or performance**
  - **AppLogSize [parameter applies to all servers]**
    - » **Maximum size of the application log (ALOG) file for a particular application**
    - » **Recommended size varies considerably depending the nature of the application for which the file is being written**
  - **AppLogLevel [parameter applies to all servers]**
    - » **Level of detail provided in the ALOG file for a particular application**
    - » **Acceptable values are 0, 1, 2, or 3**
    - » **A setting of “0” provides the most data**

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **DebugLevel [parameter applies to all servers]**
    - » Level of detail provided in the debug log file for a particular application
    - » Normally acceptable values are 0, 1, 2, or 3
    - » A setting of "0" turns off logging; a setting of "3" provides a significant amount of data
  - **DpPr\_MAX\_RETRIES [EcDpPrEM and EcDpPrDeletion parameter]**
    - » Number of retries (e.g., 30) to the Science Data Server for acquires/inserts before giving up
  - **DpPr\_WAIT\_PERIOD [EcDpPrEM and EcDpPrDeletion parameter]**
    - » Time in seconds (e.g., 120) to wait between retries to the Science Data Server

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **DpPrRM\_MAX\_RETRIES** [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter]
    - » Maximum number (e.g, 100) of attempts to allocate a computer resource
  - **DpPrRM\_RETRY\_PERIOD** [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter]
    - » Number of seconds (e.g., 120) between retries when trying to allocate a resource
  - **DpPrMaxConcurrentDPRs** [EcDpPrJobMgmt parameter]
    - » Maximum allowed jobs
    - » Three integer values (e.g., 100 100 100) are assigned to DpPrMaxConcurrentDPRs; for routine processing, on-demand processing, and reprocessing respectively

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **DpPrMinConcurrentDPRs [EcDpPrJobMgmt parameter]**
    - » **Minimum allowed jobs**
    - » **Three integer values (e.g., 0 0 0) are assigned to DpPrMaxConcurrentDPRs; for routine processing, on-demand processing, and reprocessing respectively**
    - » **NOT CURRENTLY USED**
  - **DpPrAutoSysMaxDPRs [EcDpPrJobMgmt parameter]**
    - » **Total number of jobs (e.g., 100) allowed in AutoSys**

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **DpPrDeleteFailedPGEJobs [EcDpPrJobMgmt parameter]**
    - » If TRUE, failed PGE Jobs are removed by Job Management, as necessary, when space is needed for another job that is ready to run
    - » If FALSE (the usual value), failed PGE Jobs are left in AutoSys
  - **DBConnections [EcPoConnections (includes EcPISubMgr, EcPIOdMgr, EcDpPrDeletion, EcDpPrJobMgmt and EcDpPrJobMgmtClient) parameter]**
    - » Number of connections needed by a particular application (e.g., 10 for EcPIOdMgr)
    - » Optional parameter that specifies the number of connections to maintain in the connection pool

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **SleepDelayForFailures [EcPISubMgr parameter]**
    - » Amount of time in seconds (e.g., 60) to wait before reprocessing failed notifications
    - » Sleep delay used by the failed notification thread
    - » Less frequent checking can increase speed for the other threads
  - **SleepDelayForTimers [EcPISubMgr parameter]**
    - » Amount of time in seconds (e.g., 60) the Subscription Manager should sleep between checking for expired timers
    - » Should be set to the minimum amount of time a timer will be set for at the particular DAAC (min 60 sec)
    - » Sleep delay used by the timer checking thread
    - » Less frequent checking can increase speed for the other threads

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **SleepDelayForExp [EcPIOdMgr parameter]**
    - » **Sleep delay for expiration thread in seconds (e.g., 86400)**
    - » **Should be considerably greater than the sleep delay for completion threads (SleepDelayForCmp)**
  - **SleepDelayForCmp [EcPIOdMgr parameter]**
    - » **Sleep delay for completion threads in seconds (e.g., 300)**
    - » **Should be considerably less than the sleep delay for expiration threads (SleepDelayForExp)**

# Tuning System Configuration Parameters (Cont.)



- **Tuning Parameters (Cont.)**
  - **SocketLimit [EcDpPrDeletion, EcDpPrJobMgmt, EcPIOdMgr, EcPISubMgr parameter]**
    - » **Number of connections (e.g., 200) to a server through the Hubble Space Telescope (HST) sockets middleware**
    - » **Too low a number misses connections**
    - » **Too high a number may adversely affect the memory of the server's host**

# Tuning System Configuration Parameters (Cont.)



- **When the value assigned to a parameter has been changed and saved, the modified value does not take effect until the affected server has been restarted**
- **Example**
  - **Debug level for the Subscription Manager log has been changed from “2” to “3” in the Configuration Registry**
  - **Modification does not affect the recording of data in the log until after a warm restart of the Subscription Manager (at which time the server would read the parameters in the Configuration Registry)**

# Tuning System Configuration Parameters (Cont.)



- **Production Planner and Production Monitor should work with the Resource Planner to make optimum use of processing resources**
  - Resource Planner allocates the disk partitions, CPUs, and RAM available for processing among the active modes (e.g., OPS, TS1, TS2)
  - Production Planner and Production Monitor monitor the load on the processing resources

# Tuning System Configuration Parameters (Cont.)



- **Resource Planner assigns the bulk (typically 60% - 80%) of the processing resources to the OPS mode**
  - The remainder of the processing assets are divided among the modes used for SSI&T and new version software checkout
- **The Production Planner and Production Monitor monitor the load on the processing resources to identify whether the actual load is appropriately distributed among modes**
  - They inform the Resource Planner of under- or over-use of resources as allocated

# Tuning System Configuration Parameters (Cont.)



- **Disk space allocation**
  - Disk space allocated to OPS mode is likely to be used to capacity
  - Disk space assigned to the other two modes may not fill up
- **CPU allocation**
  - There is no one-to-one mapping of CPU allocation with actual CPUs on the science processor
  - The operating system(OS) takes care of true CPU and RAM allocation
    - » Actual CPU usage during processing is limited by OS
    - » If ten CPUs have been specified for a particular mode, only ten DPRs can be running the Execute job at a given time
    - » What is really being defined is the maximum number of DPRs that will execute at a given time

# Tuning System Configuration Parameters (Cont.)



- **CPU allocation (Cont.)**
  - CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs on each science processor
    - » If monitoring indicates that the processor is underused when OPS mode is at full processing capacity, the number of CPUs allocated to OPS mode could probably be increased
    - » If the science processor is at full capacity when OPS mode is at full processing capacity (and the processor may be overworked) the number of CPUs allocated to OPS mode should be reduced
- **Random-access memory (RAM) allocation**
  - Subject to the same considerations as CPUs
  - RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor

# Tuning System Configuration Parameters (Cont.)



- **Strategies for Tuning**
  - **Section includes...**
    - » **Scenario that demonstrates how DPRs might be processed under a particular set of conditions**
    - » **Some strategies for tuning the system**
  - **The processing conditions include the following types of items:**
    - » **The total number of jobs allowed into AutoSys**
    - » **The number of CPUs available for processing**
    - » **Characteristics of the PGEs to be processed**

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing)**
  - The total number of jobs (DPRs) allowed into AutoSys is controlled by the DpPrPgeLimits table in the PDPS database
  - An example of some of the types of data maintained in the DpPrPgeLimits table is shown in the following table:

<b>computerName</b> [Virtual Computer]	<b>pgeld</b>	<b>maxConcurrent</b> [DPRs]
A	1	20
B	1	20
A	2	20
B	2	20

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing) (Cont.)**
  - Scenario assumes that each of the virtual computers (i.e., A and B) listed in the preceding table has 16 CPUs
    - » 32 CPUs total
  - Relevant PGE characteristics are shown in the table that follows:

PGE	# CPUs Used	Average Execution Time	Average Stage Time	Destage Time
1	1	5 minutes	5 minutes	5 minutes
2	1	60 minutes	5 minutes	5 minutes

- Scenario assumes that 100 DPRs of each type (i.e., PGE 1 and PGE 2 - 200 DPRs total) are ready to run and are released at once into AutoSys

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing) (Cont.)**
  - Eighty (80) DPRs enter AutoSys
  - The remaining 120 DPRs are queued, with their assignments already made:
    - » **Machine (Virtual Computer) A**
      - 20 PGE 1s start staging
      - 30 PGE 1s are queued on Machine A
      - 20 PGE 2s start staging
      - 30 PGE 2s are queued on Machine A
    - » **Machine (Virtual Computer) B**
      - 20 PGE 1s start staging
      - 30 PGE 1s are queued on Machine B
      - 20 PGE 2s start staging
      - 30 PGE 2s are queued on Machine B

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing) (Cont.)**
  - After about five (5) minutes, all 80 DPRs that were staging have finished staging and are ready for execution
    - » However, only 32 CPUs are available
  - The first 32 DPRs that ask for CPUs get them and start running
    - » Sixteen (16) on Machine A and sixteen (16) on Machine B
  - Forty-eight (48) DPRs are waiting
    - » Assuming that parameters in the Registry database are set as follows:
      - DpPrRM\_RETRY\_PERIOD = 120 seconds
      - DpPrRM\_MAX\_RETRIES = 100
    - the waiting DPRs keep trying every two minutes for up to 100 times each before timing out (after 200 min.)
    - » In this example timing out is a real possibility

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing) (Cont.)**
  - **The quick jobs complete processing after five (5) minutes, freeing up sixteen (16) CPUs**
    - » **In the current example, the sixteen (16) CPUs are subsequently occupied with about eight (8) five-minute PGEs and eight (8) 60-minute PGEs because CPUs are given randomly to whichever DPR gets back first to asking for them after waiting for the retry period (i.e., 120 seconds)**
    - » **Priorities are not used**
    - » **At first, there was a 50:50 ratio of fast:slow DPRs, now there is a 25:75 ratio of fast:slow**
    - » **After another five (5) minutes, the ratio becomes 12.5:87.5 fast:slow, so 87.5 % of the CPUs are occupied by 60-minute DPRs**

# Tuning System Configuration Parameters (Cont.)



- **Scenario (DPR Processing) (Cont.)**
  - **The 60-minute DPRs tend to dominate the CPUs**
    - » **After one (1) hour the first batch of sixteen (16) 60-minute PGEs vacates the CPUs to be replaced by eight (8) five-minute PGEs and eight (8) 60-minute PGEs, but the five-minute PGEs become extinguished again by the slow ones**
    - » **If the staging and destaging times were not the same (so the DPRs didn't have the same opportunity to hit the execution stage at the same time) the scenario would proceed differently**

# Tuning System Configuration Parameters (Cont.)



- **Strategies for Tuning the System:**
  - Limit the number of DPRs through the use of the `DpPrPgeLimitsTable`
  - Increase the declared number of CPUs for the processors to more than the actual number (overallocate CPUs)
  - Create new virtual computers (assigning CPUs on the processors to them) and assign (via the `DpPrPgeLimits` table) PGEs to run on the new virtual computers

# Tuning System Configuration Parameters (Cont.)



- **Strategies for Tuning the System (Cont.):**
  - **Limit the number of DPRs through the use of the DpPrPgeLimitsTable**
    - » In the example if the number of slow DPRs allowed into AutoSys is less than the number of CPUs, there is always a channel for the fast jobs to squeeze through
    - » The big disadvantage to this approach is that the slow jobs are also being prevented from staging
  - **Increase the declared number of CPUs for the processors to more than the actual number (overallocate CPUs)**
    - » This approach allows more of each type of PGE into the science processors
    - » The disadvantage is that it could overwhelm the science computers; however, they are kept busy

# Tuning System Configuration Parameters (Cont.)



- **Strategies for Tuning the System (Cont.):**
  - **Create new virtual computers (assigning CPUs on the processors to them) and assign (via the DpPrPgeLimits table) PGEs to run on the new virtual computers**
    - » **This approach is another way to guarantee bandwidth (CPUs) to PGEs**
    - » **The disadvantage of this approach is that some CPUs could remain idle, not being seen by one of the virtual computers**
    - » **In the past, there may have also been some code problems with supporting this, but those difficulties should have been resolved**

# Tuning System Configuration Parameters (Cont.)



- **Strategies for Tuning (Cont.)**
  - **Probably some combination of the first two strategies is best**
    - » **Increase the number of declared CPUs to be more than the total number of slow jobs allowed into AutoSys, always leaving some CPUs for a channel of fast jobs**
    - » **The total number of faster-moving jobs should be increased to make sure that there is always be a queue of them available to get their channel occupied**
  - **The staging and destaging times have to be accounted for and this could change things in terms of using the DpPrPgeLimits table and the number of CPUs per processor to tune the job flow**
  - **It is important to perform regular garbage collection on all of the virtual computers**

# Troubleshooting Resource Planning Problems



- **Troubleshooting**

- **Process of identifying the source of problems on the basis of observed trouble symptoms**
- **Problems with Resource Planning can usually be traced to either some part of the Planning Subsystem or the ECS infrastructure**
  - » **Resource Planning does not have interfaces with many other subsystems**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery**

- Each request that crosses a client/server boundary is assigned a system-unique identifier referred to as an RPC ID
- The RPC ID facilitates the automatic fault recovery events that occur whenever there is a client or server failure
- As a request propagates through the system, each associated client/server exchange is assigned a unique RPC ID
  - » The RPC ID for each interaction is derived from the previous RPC ID received by the client for the request; consequently, all RPC IDs associated with a given request have a common portion that relates the various client/server calls to one another
  - » Given the previous RPC ID, clients consistently reproduce the same RPC ID that was submitted to the server on the subsequent event

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery (Cont.)**

- **The concept of reproducible RPC IDs is central to the ECS fault recovery capability**
  - » **When requests are retried from client to server, they are always submitted with the same RPC ID that was used in the original submission of the request, even if either client or server has crashed between retries**
- **The RPC ID is also central to the check-pointing aspect of fault recovery**
  - » **As requests arrive at fault recovery-enabled servers, they are recorded in a persistent store (typically a database), tagged with the RPC ID**
  - » **As the request is serviced, check-pointing state information may be updated in the persistent store, up to and including the request's completion status**
  - » **This allows the servers to resume servicing from the last check-pointed state, particularly upon resubmission from a client**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Fault Handling**
  - **Failure events are classified according to the following three severity levels:**
    - » **Fatal error**
    - » **Retry error**
    - » **Warning**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Fault Handling (Cont.)**
  - **Fatal error is returned when a request cannot be serviced, even with operator intervention**
    - » **For example, if a request is made to distribute data via ftp to a non-existent host, the request is failed**
  - **Retry error is a potentially recoverable error**
    - » **Normally, a retry error would be returned to the client only when the server cannot recover from the error automatically**
    - » **A retry error may require operator assistance**
    - » **For example, entering a new name for a resource reservation request after being notified that there was a previously existing request of the same name**
  - **Warning is provided when operations can proceed but an unexpected circumstance was detected**
    - » **For example, if a client requests removal of a file but the file does not exist**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Fault Handling (Cont.)**
  - **Transient errors (such as network errors) are always retry errors**
    - » **In general, clients and servers that experience transient retry errors first attempt to recover by retrying the operation automatically**
    - » **One special case of this is “rebinding,” which refers to the process by which a client automatically attempts to re-establish communication with a server in the event communication is disrupted**
    - » **The disruption may be caused by transient network failure, or by the server crashing or being brought down**
    - » **In any case, the client automatically attempts to reconnect to the server for a configurable period of time on a client-by-client basis**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Fault Handling (Cont.)**
  - ECS processes encountering an error or receiving an error from a server request can either pass the error back to a higher-level client or present it to the operator for operator intervention

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Client Crash and Restart**
  - In general when a client crashes, the server continues to service the requests that were in process at the time of the client's crash
  - When a client restarts in the ECS system, it sends a restart notification to each server with which it interacts
    - » Clients notify servers that they have come up either "cold" or "warm"
    - » Generally, the notification temperature sent to the server matches the temperature at which the client process is restarted

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Client Crash and Restart (Cont.)**
  - **Default server behavior in response to “warm” startup notification from a client:**
    - » **Outstanding requests for the restarted clients remain available in the persistent store**
    - » **The outstanding requests may be resubmitted by the client, and are serviced to completion upon resubmission**
    - » **Associated resources are left allocated until the requests are completed**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Client Crash and Restart (Cont.)**
  - **Default server behavior in response to “cold” startup notification from a client:**
    - » **All outstanding requests for the restarted client are cancelled**
    - » **If the client resubmits any cancelled request using the same RPC ID (e.g., by pressing the Retry button from an operator GUI), it is failed with a fatal error due to the client cold startup notification**
    - » **Any resources associated with the cancelled requests are released and reclaimed by the system**

# Troubleshooting Resource Planning Problems (Cont.)



- **Fault Recovery: Server Crash and Restart**
  - When a server crashes, clients cannot continue to submit requests for processing
  - Synchronous requests in progress result in a Distributed Computing Environment (DCE) exception being thrown back to the client process, which enters a rebinding failure recovery mode (as previously mentioned)
  - Attempts to submit requests while the server is down result in the client blocking until a communication timeout has been reached
  - Although DCE has been replaced by socket-based library calls (i.e., CCS Middleware), the DCE exception code is handled by the CCS Middleware

# Troubleshooting Resource Planning Problems (Cont.)



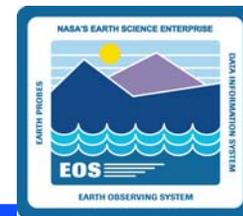
- **Fault Recovery: Server Crash and Restart (Cont.)**
  - **When a server restarts, it may perform various resynchronization activities in order to recover from an unexpected termination**
  - **In the event of a server cold start or cold restart, the server typically cancels all outstanding requests and reclaims all associated resources**
  - **In general, existing request queues are retained for warm restarts and cleared for cold starts or cold restarts**

# Troubleshooting Resource Planning Problems (Cont.)



- **Troubleshooting table**
  - **Describes actions to be taken in response to some common Resource Planning problems**
  - **If the problem cannot be identified and fixed without help within a reasonable period of time, call the help desk and submit a trouble ticket in accordance with site Problem Management policy**

# Troubleshooting Resource Planning Problems (Cont.)



Symptom	Response
Unable to log in to the Planning Subsystem host (e.g., g0pls01).	Check with the Operations Controller/System Administrator to ensure that the host is "up."
GUI not displayed when the start-up script has been properly invoked.	Ensure that the DISPLAY variable was set properly. [For detailed instructions refer to the procedure for <b>Launching Resource Planning Applications Using UNIX Commands</b> (previous section of this lesson).]
Error message indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID.	<ol style="list-style-type: none"> <li>1. Use another Application ID if working in a different mode from the person using the selected Application ID.</li> <li>2. If working in the same mode as the other user, coordinate use of Planning applications with the other user and/or the System Administrator.</li> </ol> [For detailed instructions refer to the procedure for <b>Launching Resource Planning Applications Using UNIX Commands</b> (previous section of this lesson).]
Error message associated with the Resource Editor.	Refer to Table 6, Resource Editor User Messages (adapted from the corresponding table in 609-EMD-001, <i>Release 7 Operations Tools Manual for the EMD Project</i> ).
Error message associated with the Resource Scheduler.	Refer to Table 7, Resource Scheduler User Messages (adapted from the corresponding table in 609-EMD-001, <i>Release 7 Operations Tools Manual for the EMD Project</i> ).

# Troubleshooting Resource Planning Problems (Cont.)



Symptom	Response
Other problems.	Check the log files (e.g., EcPIRpre.ALOG, EcPIRpsI.ALOG, EcPIRprM.ALOG) in the /usr/ecs/MODE/CUSTOM/logs directory for error messages. [For detailed instructions refer to the procedure for <b>Checking Log Files</b> (subsequent section of this lesson).]

# Troubleshooting Procedures



- **The following procedures for correcting Resource Planning problems are provided in this section:**
  - **Checking Log Files**
  - **Checking Database Connections**

# Checking Log Files



- **Log files can provide indications of the following types of problems:**
  - **Resource Planning-related problems**
  - **Communication problems**
  - **Database problems**
  - **Lack of disk space**

# Checking Log Files (Cont.)



- **Procedure**

- **Access a terminal window logged in to the appropriate host**
- **Change directory to the directory containing the resource planning log files**
  - » **/usr/ecs/MODE/CUSTOM/logs**
- **Review log file to identify problems**
  - » **EcPIRpRe.ALOG**
  - » **EcPIRpReDebug.log**
  - » **EcPIRpSi.ALOG**
  - » **EcPIRpSiDebug.log**
- **Respond to problems**

# Checking Database Connections



- **PDPS database**
  - Repository of data concerning planning and processing
  - If applications are unable to connect to the database, the data cannot be retrieved or displayed on the GUI
  - Checking the database connections is a logical step in trying to isolate the following types of problems:
    - » GUI does not display data
    - » Display does not refresh

# Checking Database Connections



- **Procedure**

- **Submit a request to the Database Administrator to identify the values for the following parameters:**
  - » **DBName**
  - » **DBServer**
  - » **DBMaxConnections**
- **Use the interactive structured query language (isql) sp\_who command to obtain a list of actual connections**
- **Use the isql sp\_configure command to obtain a list of the number of connections for which the database has been configured**
- **Compare the number of actual connections (results of sp\_who) with the number of connections for which the database has been configured (results of sp\_configure "user connections")**
- **Notify the Database Administrator of problems**