

# **Using SAS® Client Applications with SAS® Grid Manager**

**Course Notes**

*Using SAS® Client Applications with SAS® Grid Manager Course Notes* was developed by Martha Hays, Christine Riddiough, and Scott Vodicka. Editing and production support was provided by the Curriculum Development and Support Department.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.

### **Using SAS® Client Applications with SAS® Grid Manager Course Notes**

Copyright © 2016 SAS Institute Inc. Cary, NC, USA. All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

---

Book code E70656, course code LWPCCCL61/PCCL61, prepared date 08Apr2016. LWPCCCL61\_002

ISBN 978-1-62960-314-8

## Table of Contents

Course Description .....	v
Prerequisites .....	vi
<b>Chapter 1     Introduction.....</b>	<b>1-1</b>
1.1   Overview.....	1-3
1.2   SAS Grid Manager Architecture.....	1-10
1.3   Understanding Grid Processing .....	1-15
1.4   Solutions .....	1-20
<b>Chapter 2     Monitoring Grid Execution .....</b>	<b>2-1</b>
2.1   Platform RTM .....	2-3
Demonstration: Using Platform RTM.....	2-8
Exercises.....	2-11
2.2   SAS Management Console .....	2-12
Exercises.....	2-17
2.3   Command Line Tools.....	2-19
Exercises.....	2-23
2.4   Solutions .....	2-24
Solutions to Exercises .....	2-24
<b>Chapter 3     Using SAS® Management Console and Schedule Manager with a SAS® Grid .....</b>	<b>3-1</b>
3.1   Scheduling on a SAS Grid .....	3-3
Demonstration: Creating and Scheduling a Flow .....	3-14
Exercises.....	3-20
3.2   Solutions .....	3-21
Solutions to Exercises .....	3-21
<b>Chapter 4     SAS® on the Grid – Working with Code.....</b>	<b>4-1</b>
4.1   SAS Grid Submission in the SAS Windowing Environment.....	4-3
Demonstration: Submitting to the Grid in the SAS Windowing Environment .....	4-24

Exercises.....	4-26
4.2 SAS Grid Enablement with the SCAPROC Procedure .....	4-27
Demonstration: Using the SCAPROC Procedure .....	4-35
Exercises.....	4-38
4.3 SAS Grid Submission Using the SASGSUB Utility .....	4-39
Demonstration: Using SASGSUB .....	4-46
Exercises.....	4-48
4.4 Solutions .....	4-49
Solutions to Exercises .....	4-49
<b>Chapter 5     SAS® Client Applications on the Grid.....</b>	<b>5-1</b>
5.1 SAS Clients and SAS Workspace Servers .....	5-3
5.2 SAS Data Integration Studio with the SAS Grid.....	5-8
Exercises.....	5-14
5.3 SAS Enterprise Guide on the SAS Grid .....	5-17
Demonstration: Using SAS Enterprise Guide on the Grid.....	5-24
Exercises.....	5-29
5.4 SAS Add-In for Microsoft Office Running a SAS Stored Process on the SAS Grid.....	5-30
Exercises.....	5-32
5.5 SAS Enterprise Miner on the SAS Grid .....	5-34
Exercises.....	5-36
5.6 Solutions .....	5-37
Solutions to Exercises .....	5-37
<b>Appendix A   Additional Information.....</b>	<b>A-1</b>
A.1 SAS Grid Resources .....	A-3
A.2 Using SAS/CONNECT.....	A-12
Demonstration: Using SAS/CONNECT .....	A-17

## Course Description

This course teaches the architecture of a SAS grid environment, how to submit SAS workload to the grid from a variety of different SAS products and user interfaces, and how to convert existing SAS programs to run on the grid. The course includes hands-on exercises that use SAS applications with SAS Grid Manager. This course is also included in [SAS Grid Manager service offerings](#), designed to help you build a grid environment that's modern, flexible, and forward-looking.

### To learn more...



For information about other courses in the curriculum, contact the SAS Education Division at 1-800-333-7660, or send e-mail to [training@sas.com](mailto:training@sas.com). You can also find this information on the web at <http://support.sas.com/training/> as well as in the Training Course Catalog.



For a list of other SAS books that relate to the topics covered in this course notes, USA customers can contact the SAS Publishing Department at 1-800-727-3228 or send e-mail to [sasbook@sas.com](mailto:sasbook@sas.com). Customers outside the USA, please contact your local SAS office.

Also, see the SAS Bookstore on the web at <http://support.sas.com/publishing/> for a complete list of books and a convenient order form.

## Prerequisites

Before attending this class, you should be familiar with the functionality and components of the SAS products that you will use on the grid. You can gain this knowledge by completing the Management Overview: Exploring the Platform for SAS Business Analytics course.

# Chapter 1      Introduction

<b>1.1</b>	<b>Overview.....</b>	<b>1-3</b>
<b>1.2</b>	<b>SAS Grid Manager Architecture.....</b>	<b>1-10</b>
<b>1.3</b>	<b>Understanding Grid Processing .....</b>	<b>1-15</b>
<b>1.4</b>	<b>Solutions .....</b>	<b>1-20</b>
	Solutions to Student Activities.....	1-20



## Course Goals

- There are many ways to accomplish a single task in SAS.
- The goal of this course is **not** to teach you all of the options for each of the different client applications.
- The goal is to simplify and demonstrate exploiting SAS Grid Manager from a variety of SAS client applications.

2

## 1.1 Overview

---

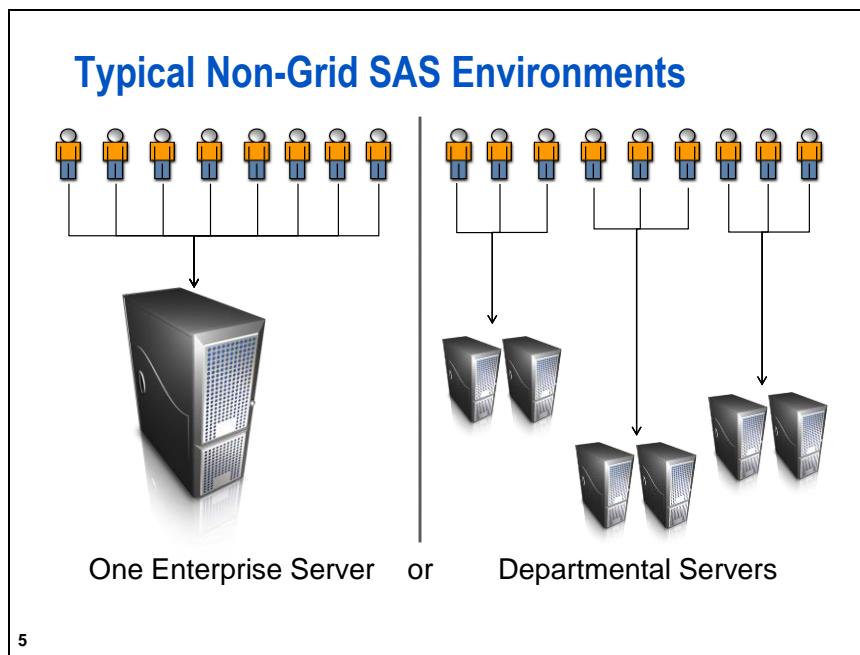
### Objectives

- Describe SAS Grid Computing.
- Define the SAS Grid Computing terminology.

4



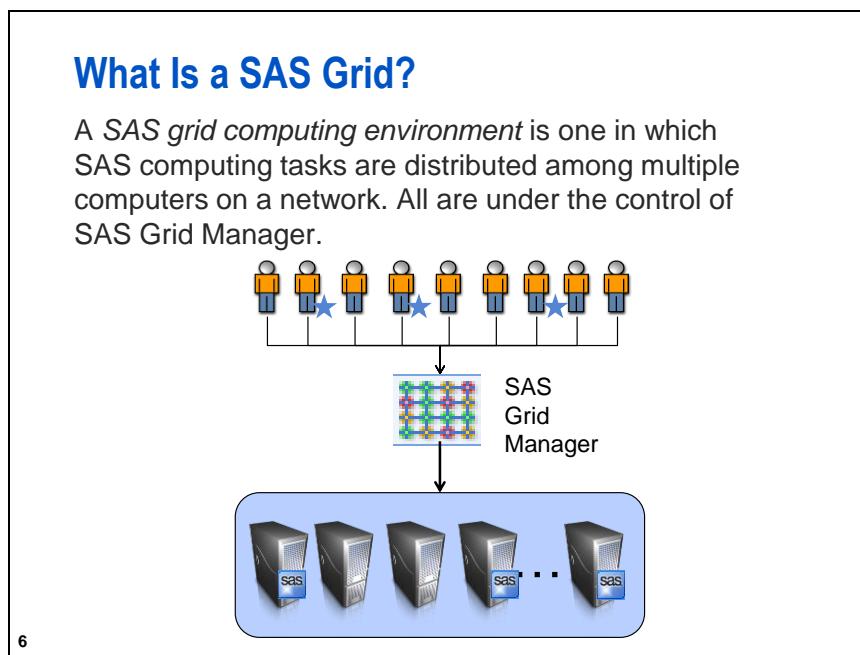
Two additional courses address other topics related to SAS Grid Computing. SAS® Enterprise Scheduling with Platform Suite for SAS® discusses many aspects of scheduling, which are briefly highlighted in a later chapter of this course. SAS® Grid Manager Administration provides detailed information about administering the grid.



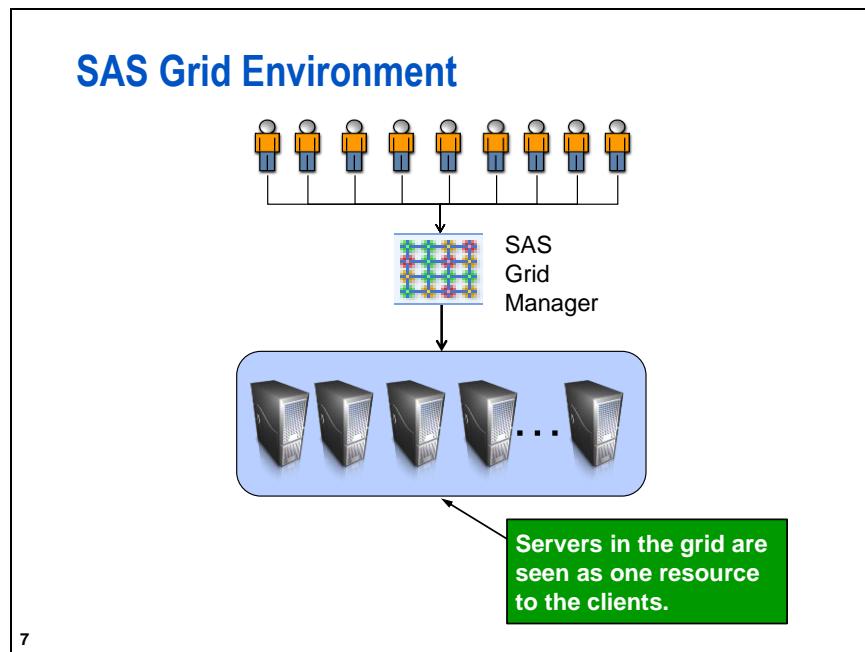
Before using SAS Grid Manager, many customers set up their SAS environments in two typical manners.

- One setup is to have a large enterprise server where all SAS users submit jobs.
- Another is to set up sets of departmental servers where SAS users submit jobs to only their department's server.

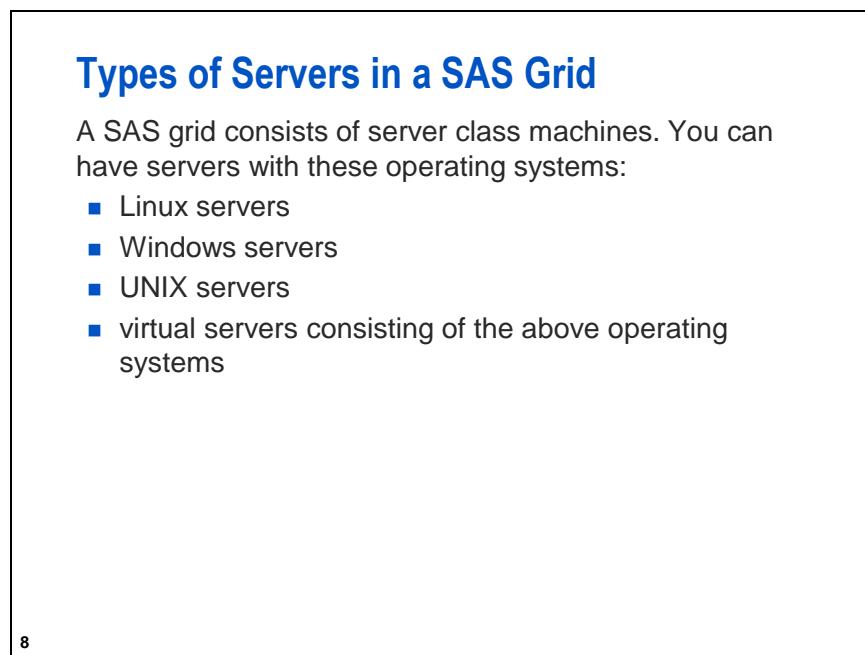
The issues that you might encounter are that all users compete for resources on the servers that they are assigned to use. If SAS programs or other processes run on your server, you have no choice but to run your SAS program on your server. Consequently, you add to the competition for computing resources although there might be other servers that are under-used.



This definition is short but descriptive, and gives a hint of what you can find “under the hood” of SAS Grid Manager. As depicted in the slide, users submit their SAS programs to SAS Grid Manager, and SAS Grid Manager distributes the SAS programs among the servers in the grid.



In a SAS grid environment, a set of computer servers are grouped together and seen from the users' point of view as one large resource of computing services. The users are no longer required to submit their SAS programs to a specific server. SAS Grid Manager finds the best available server for the SAS request when it is made.



The computer hardware used as the servers in a grid are typically server-class machines. These machines have fast processors, a significant amount of memory, and the ability to handle high data-traffic levels. The use of commodity server hardware is a popular choice for grid servers.

## Benefits of Using SAS Grid Manager

- Centralized and Shared Management
  - Administrators create, manage, and monitor grid configuration from a central control point.
  - Groups, users, and job type priorities are defined by administrators.
  - SAS Grid Manager de-couples SAS applications and programs from the infrastructure.

When you move from a non-grid environment to a SAS Grid Manager environment, there are many benefits. As a SAS administrator, you know a central location where you build your grid environment, manage the use of the grid, and make changes to the grid configuration over time from that central location.

Now, with a SAS Grid Manager environment, you can set it up to meet the needs of your enterprise, and change as needs require.

## Benefits of Using SAS Grid Manager

- Business Continuity
  - SAS Grid Manager delivers a highly available environment.
    - Multiple servers cover hardware failures.
    - The architecture of the grid environment provides a highly available platform.
    - Important services can be set up to be highly available.
      - » ensures that SAS applications and services are highly available\*
  - The completion of important jobs is ensured even in the event of a hardware failure.

10

*continued...*

## Benefits of Using SAS Grid Manager

- Increased Flexibility
  - The grid architecture enables servers to be easily taken offline for service.
  - Servers can be allocated on the fly or automatically.
- Efficient Use of Resources
  - Administrator-defined policies enable the control of access to resources based on the following:
    - group membership
    - time of day
    - “busyness” of the servers
    - using prioritization
  - Processing is spread among the servers.
    - prevents one server from becoming overloaded

11

*continued...*

## Benefits of Using SAS Grid Manager

- Accelerated Results
  - Jobs are routed to the least busy server to enable faster execution.
  - Servers are kept from being overloaded, which enables them to run jobs efficiently.
  - SAS programs can be split into parallel segments to run across multiple servers in parallel.
  - Resource usage is more uniform across servers.

12

## SAS Grid Manager Feature: Multi-User Workload Balancing

- The workload submitted by multiple users is spread out among the available machines.
- Balancing ensures that no machine becomes overloaded.
- Jobs are put in a waiting queue if necessary.
- Priorities can be assigned to users or applications.
- Preemption rules can be set for users or applications.
- The overall throughput of jobs is increased.

13

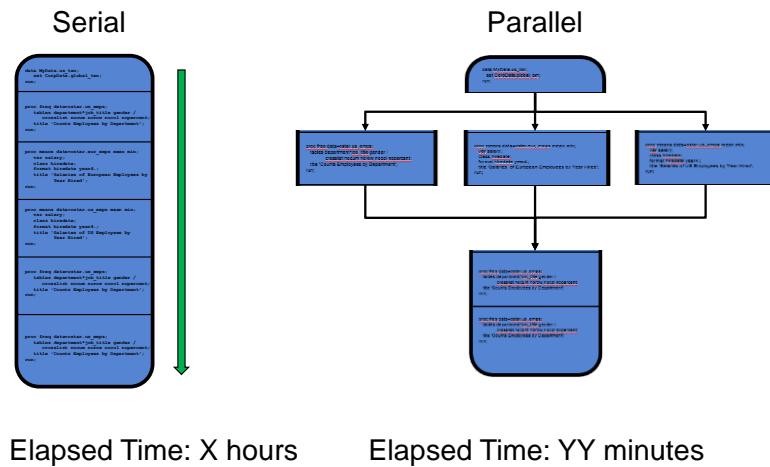
## SAS Grid Manager Feature: Parallelized Workload Balancing

*Parallelized workload balancing* can be described as follows:

- taking a SAS program with many independent steps and dividing it into sections
- adding code to enable the independent sections to run in parallel on multiple servers at the same time
- significantly reducing the run time of the SAS program

14

## Serial versus Parallelized SAS Program



15

In a SAS environment, SAS programs are run as one serial program. You can think of the SAS program as running the SAS statements one after the other from top to bottom. In this case, even if you have segments that are independent, they must wait until all the steps above are completed before the programs can run.

In a SAS Grid Manager environment, your SAS programs that have independent steps can be run in parallel across the servers in the grid. This greatly reduces the elapsed clock time that it takes your SAS programs to run. Depending on the circumstances, you might be able to take a current SAS program and reduce its processing time from hours to minutes.

The key point is that you can run independent sections of your SAS program in parallel, not only in threads on one server but also in parallel on multiple servers. Many times, this significantly decreases the running time of your SAS program.

## 1.01 Multiple Choice Poll

Which of the following are features of SAS Grid Manager?

- a. ability to make SAS servers highly available
- b. manages many simultaneous requests for SAS grid processing
- c. enables SAS programmers to run sections of code in parallel
- d. all of the above

16

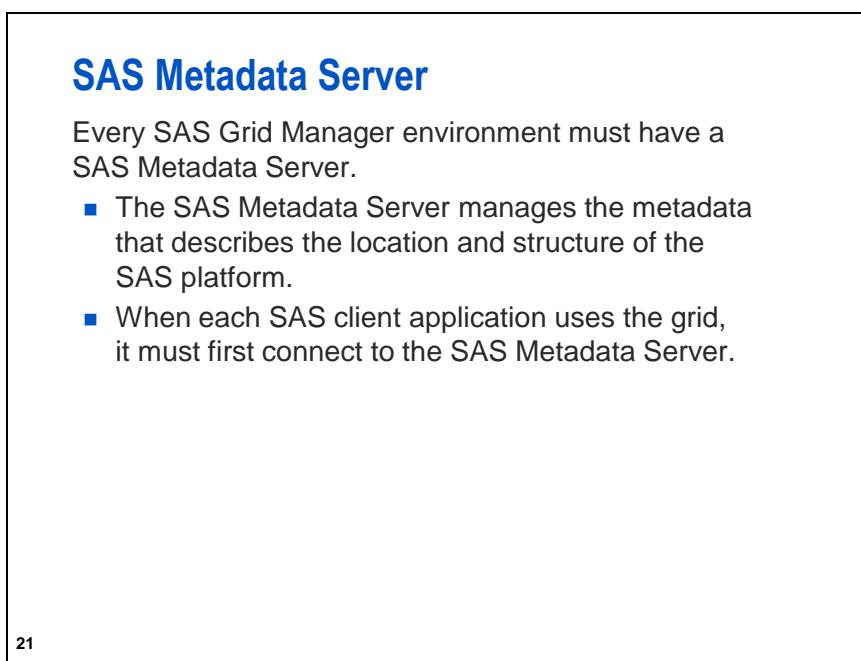
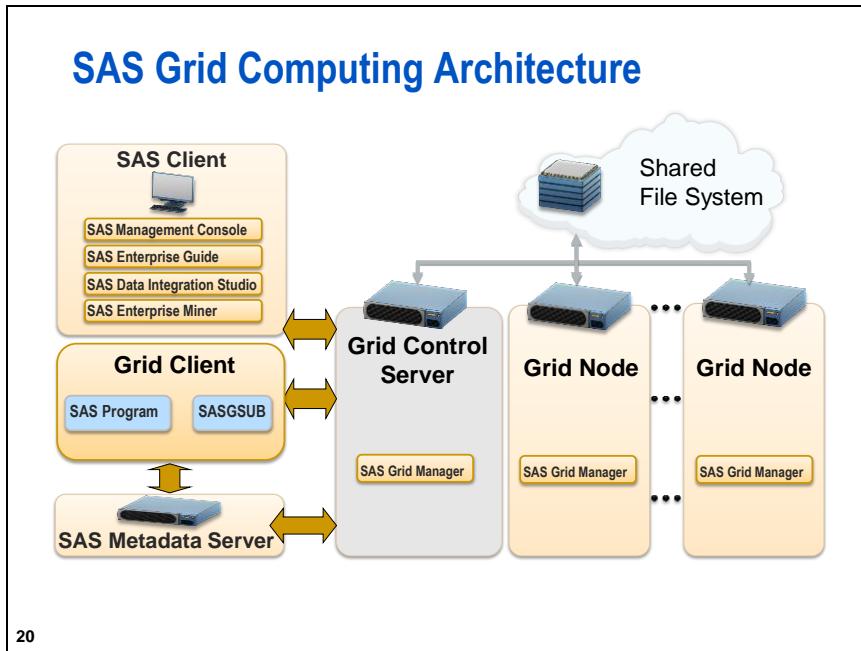
## 1.2 SAS Grid Manager Architecture

---

### Objectives

- Describe SAS Grid Manager architecture.
- Define the SAS Grid Manager components.
- Explain the purpose of the SAS Grid Manager components.

19



## SAS Grid Manager: SAS Grid Control Server

The SAS Grid Control Server controls the distribution of jobs to the grid.

- All SAS grid processing requests are sent to the SAS Grid Control Server.
- The SAS Grid Control Server stores SAS grid processing requests in queues.
- The SAS Grid Control Server distributes SAS grid processing requests among the servers in the grid based on policies set up by the administrator and to the least busy server that matches the policies.

22

## SAS Grid Manager: SAS Grid Node

A SAS *Grid node* is a server that is capable of running the jobs that are submitted to the grid.

- Nodes can be added or removed as business needs require. This is referred to as *horizontal scaling*.
- The SAS Grid nodes run SAS and the SAS code sent along with the request.
- The SAS Grid nodes perform the work for the grid.
- The SAS Grid nodes return the results and log to the requesting client application.

23

## SAS Grid Manager: Grid Client

A *grid client* is a client computer with a component of SAS Grid manager installed and configured that lets the client submit requests to the grid. The grid client can do the following:

- submit SAS programs to the grid from the SAS Foundation (using the *SAS windowing environment*, with the Editor, Log, and Output windows)
- run the SAS Grid Manager client utility to submit batch processes to the grid
  - Now, in SAS 9.4, it can start SAS on a UNIX grid in interactive mode.
  - Both line mode and X Windows mode are available.

24

## SAS Grid Manager: SAS Client

A *SAS client* is a computer without any SAS Grid Manager components installed on it.

- Some of the SAS client applications are
  - SAS Enterprise Guide
  - SAS Data Integration Studio
  - SAS Enterprise Miner.
- SAS clients generate SAS code to send to the grid to run.

25

## 1.02 Poll

A SAS Grid Manager environment does not need a SAS Metadata Server.

- True
- False

26

## 1.03 Poll

A SAS Grid Manager environment can have many SAS Grid nodes.

- True
- False

28

## 1.04 Multiple Choice Poll

When grid clients submit processing to a SAS grid, where are the requests sent?

- a. SAS Grid node
- b. SAS Metadata Server
- c. SAS Grid Control Server
- d. SAS Grid Manager Server

30

# 1.3 Understanding Grid Processing

---

## Objectives

- Describe how SAS client applications submit processing to the grid.
- List the steps that start SAS processes on the grid.
- At a high level, describe how SAS distributes requests for running SAS on the grid.

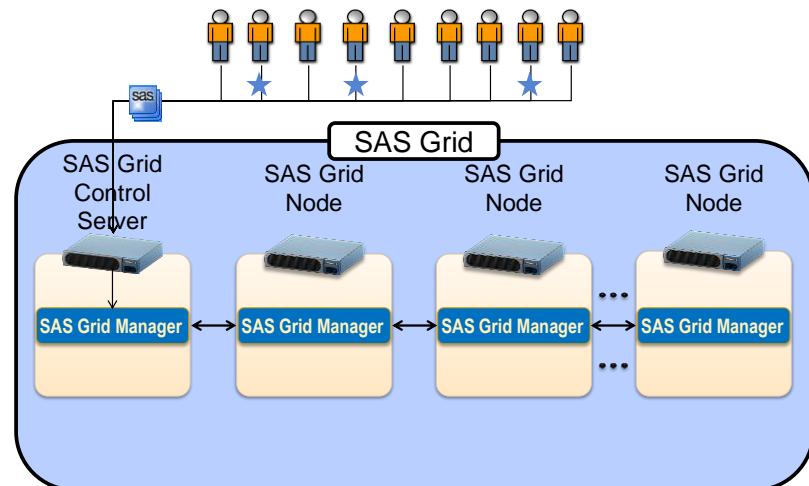
33

## SAS Client Applications: Leveraging the Grid

- SAS client applications are interface applications that generate SAS code to be sent to a SAS server for execution.
- SAS Grid Manager receives, schedules, and distributes SAS solution and program requests for remote SAS servers across a pool of managed compute resources.
- SAS client applications can redirect their requests for the SAS servers to the SAS Grid Manager.
- In a SAS Grid Manager environment, the servers that process the SAS code run on the grid nodes.

34

## SAS Grid Session Request

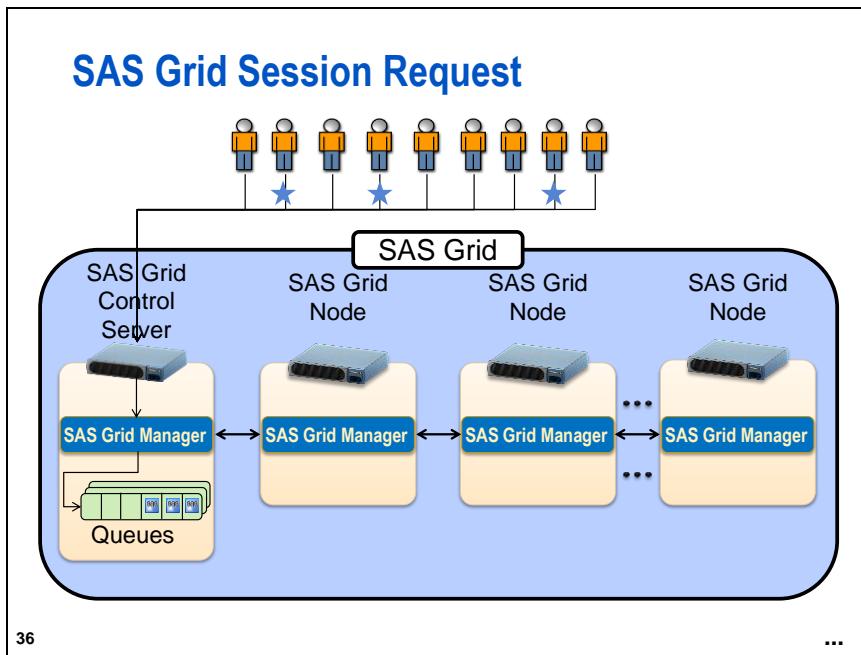


35

...

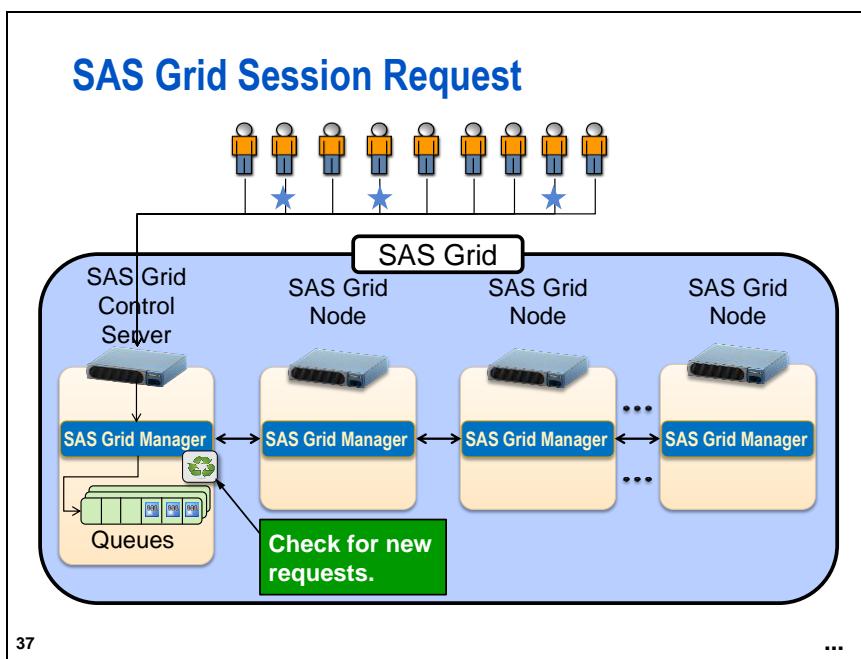
Using the various SAS applications, users submit requests for SAS processing on the grid. These requests all go to the SAS Grid Control Server. The result of the requests for SAS processing on the grid is that each user will have a SAS Grid Session running on a SAS Grid Node.

For example, a SAS Enterprise Guide user who is set up to use their SAS Grid Manager Environment opens a project and clicks the Run button. Because their Enterprise Guide is set up to use SAS Servers on the Grid, the request to run the project causes a request to go to the SAS Grid Control Server for a SAS Server, which runs the code for the project.

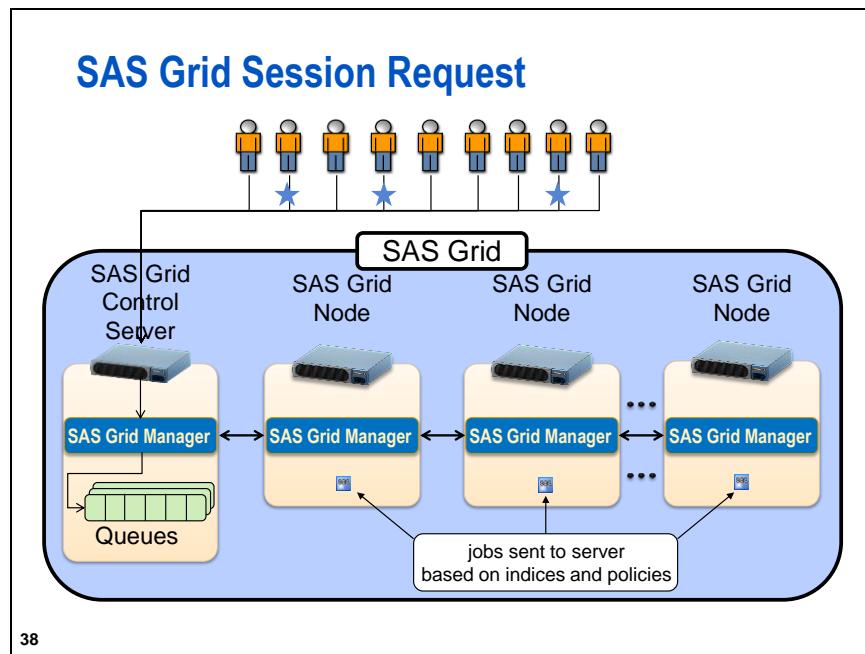


After the requests reach the SAS Grid Server, which is very quick, the SAS Grid Control Server routes the requests to a queue that was set up by the SAS Administrator. The majority of the time, users of the SAS client applications do not have the option of choosing a queue. Your SAS Administrators hide this detail in the configuration of the grid.

Having all of the requests go to one or more queues enables SAS Grid Manager to handle many simultaneous requests.



SAS Grid Manager Components on the SAS Grid Control Server periodically checks the set of queues for requests for SAS processing on the grid. Remember that a request for SAS processing on the grid is a request for a SAS Grid Session on one of the SAS Grid Nodes.



The SAS Grid Manager Components on the SAS Grid Control Server find the “least busy” server from the set of servers the user is allowed to use to start a SAS Grid Session on. After the SAS Grid Session starts, it is connected with the client and will perform the SAS processing.

## 1.05 Poll

A SAS Grid Manager environment can have only one queue.

- True
- False

## 1.06 Poll

The SAS administrator configures policies to determine which servers you can use in the grid and how they are selected.

- True
- False

## 1.4 Solutions

---

### Solutions to Student Activities

#### 1.01 Multiple Choice Poll – Correct Answer

Which of the following are features of SAS Grid Manager?

- a. ability to make SAS servers highly available
- b. manages many simultaneous requests for SAS grid processing
- c. enables SAS programmers to run sections of code in parallel
- d.all of the above

SAS Grid Manager enables SAS servers to be set up to be highly available, handles many simultaneous requests for SAS grid processing, and enables sections of code to run in parallel.

17

#### 1.02 Poll – Correct Answer

A SAS Grid Manager environment does not need a SAS Metadata Server.

- True
- False

Every SAS Grid Manager environment must have an installed SAS Metadata Server that is configured and running. The SAS Metadata Server controls access to the SAS grid.

27

### 1.03 Poll – Correct Answer

A SAS Grid Manager environment can have many SAS Grid nodes.

- True
- False

SAS Grid Manager environments can have many SAS Grid nodes, with the SAS processing spread across them.

29

### 1.04 Multiple Choice Poll – Correct Answer

When grid clients submit processing to a SAS grid, where are the requests sent?

- a. SAS Grid node
- b. SAS Metadata Server
- c. SAS Grid Control Server
- d. SAS Grid Manager Server

All requests for SAS processing on a SAS grid are sent to the SAS Grid Control Server, which distributes the requests among the grid nodes.

31

## 1.05 Poll – Correct Answer

A SAS Grid Manager environment can have only one queue.

- True
- False

Your SAS grid administrator sets up several queues. He or she maps the users of the SAS client applications to the queues that are configured to support each group of users.

40

## 1.06 Poll – Correct Answer

The SAS administrator configures policies to determine which servers you can use in the grid and how they are selected.

- True
- False

42

# Chapter 2 Monitoring Grid Execution

<b>2.1 Platform RTM .....</b>	<b>2-3</b>
Demonstration: Using Platform RTM .....	2-8
Exercises .....	2-11
<b>2.2 SAS Management Console.....</b>	<b>2-12</b>
Exercises .....	2-17
<b>2.3 Command Line Tools.....</b>	<b>2-19</b>
Exercises .....	2-23
<b>2.4 Solutions .....</b>	<b>2-24</b>
Solutions to Exercises .....	2-24
Solutions to Student Activities.....	2-30



## 2.1 Platform RTM

### Objectives

- Describe grid monitoring tools.
- Explain Platform RTM for SAS.

3



### Monitoring SAS Grid Execution

Three tools are available for monitoring grid execution.

- Platform RTM for SAS
- SAS Management Console with the SAS Grid Manager plug-in
- Command line tools

4

## Monitoring SAS Grid Execution

- Platform RTM for SAS
  - web-based application
  - includes all monitoring covered by SAS Management Console
  - alerts and high-availability controls
  - grid management and configuration tools

5

Only Administrators have the capability to make changes to the SAS Grid Environment. Users who are not administrators do not see the Configuration section of Platform RTM for SAS.

## Monitoring SAS Grid Execution

- Platform RTM for SAS enables monitoring of jobs by
  - entire grid
  - host
  - queue.
- Drill down into subset of jobs for
  - entire grid
  - selected host
  - selected queue.
- View the details of jobs.
  - job ID

6

**Platform RTM for SAS**

Platform RTM - Batch Jobs Details - Windows Internet Explorer  
http://localhost:act/plugins/grid/grid\_jobs.php?clusterid=0&user=-1&status=RUNNING&efficiency=-1&queue=-1&ex\_host=

config graphs alerts **grid** HA

Grid -> View Job Listing

Batch Job Filters

Cluster:	All	User:	All	UGroup:	All	Status:	RUNNING
Queue:	All	Host:	All	HGroup:	All	Records:	30
JobID:		Search:		<input checked="" type="checkbox"/> Dynamic <b>go</b> clear			

<< Previous Showing Rows 1 to 3 of 3 [1]

Job	JobID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time
4421	SASGrid:94692		RUNNING	2	\asas	-	-	11-30 16:18:19
4422	SASGrid:94692		RUNNING	2	\asas	-	-	11-30 16:18:19
4423	SASGrid:94692		RUNNING	2	\asas	-	-	11-30 16:19:00

<< Previous Showing Rows 1 to 3 of 3 [1]

Warning Efficiency	Alarm Efficiency	Flapping	Dependencies	Invalid Dependencies
--------------------	------------------	----------	--------------	----------------------

Last Refresh : 4:19:13 pm

7

## Viewing Grid Jobs in Platform RTM for SAS

- Platform RTM for SAS provides filters for selecting jobs to view.
- Filters enable you to subset a list of jobs by
  - user or user group
  - status (running, finished, suspended, and so on)
  - hosts
  - queue.

8

## Platform RTM for SAS

**Grid** ⇒ **Cluster** displays information about the entire grid.

**Cluster** = the set of servers that are grouped together as one unit in your grid.

The screenshot shows the 'Cluster' dashboard under the 'Grid' tab. A red circle highlights the 'Cluster' link in the left sidebar. A red box highlights the table in the main area, which displays the following data:

Actions	Cluster Name	Cluster Status	Master Status	PAU	Collect Status	CPU %	Slot %	Efc %	Total CPU's	Host Slots
	sas_cluster	Ok/Ok	Ok/Ok	P	Up	100%	0%	N/A	2	5
<b>Totals</b>										
2 5										

9

To view information about all servers in your grid, you use the Platform RTM for SAS Grid tab. Then select the **Cluster** link in the menu on the left side (under Dashboards). In this view, you can see the number of servers (referred to as hosts), CPUs, and so on, in your grid.

In a SAS Grid Manager environment, the set of all servers that make up your grid are referred to collectively as a *cluster* by Platform RTM for SAS.

## Platform RTM for SAS

**Grid** ⇒ **Management** ⇒ **By Queue**

displays information about jobs per queue.

The screenshot shows the 'By Queue' dashboard under the 'Management' tab. A red circle highlights the 'By Queue' link in the left sidebar. A red box highlights the table in the main area, which displays the following data:

Actions	Queue Name	Cluster Name	Priority	Status Reason	Max Slots	Num Slots**	Run Slots	Pend Slots	Susper Slots
	idle	cluster1	20	Open:Active	5	0	0	0	0
	night	cluster1	40	Open:Inactive	5	0	0	0	0
	chkpnt_rerun_queue	cluster1	40	Open:Active	5	0	0	0	0
	short	cluster1	35	Open:Active	5	0	0	0	0
	license	cluster1	33	Open:Active	5	0	0	0	0
	normal	cluster1	30	Open:Active	5	0	0	0	0
	owners	cluster1	43	Open:Active	5	0	0	0	0
	unicodecmd	cluster1	30	Open:Active	5	0	0	0	0
	priority	cluster1	43	Open:Active	5	0	0	0	0

10

Within Platform RTM, you find information about grid execution, specifically job status. You can look at jobs grouped by several categories. This view shows information about each queue including information on the number of active jobs associated with each queue. (See in the slide above – each queue can have jobs in these states: Run, Pend, Suspend.) Notice at the top of the page that the filter for the **Status** field is initially set to **Running**. To see the status of completed jobs, change the filter to **Finished**.

## 2.01 Multiple Answer Poll

Which of the following does Platform RTM for SAS enable you to do?

- a. view job details
- b. view queue information
- c. subset job information based on status
- d. subset job information based on user



## Using Platform RTM

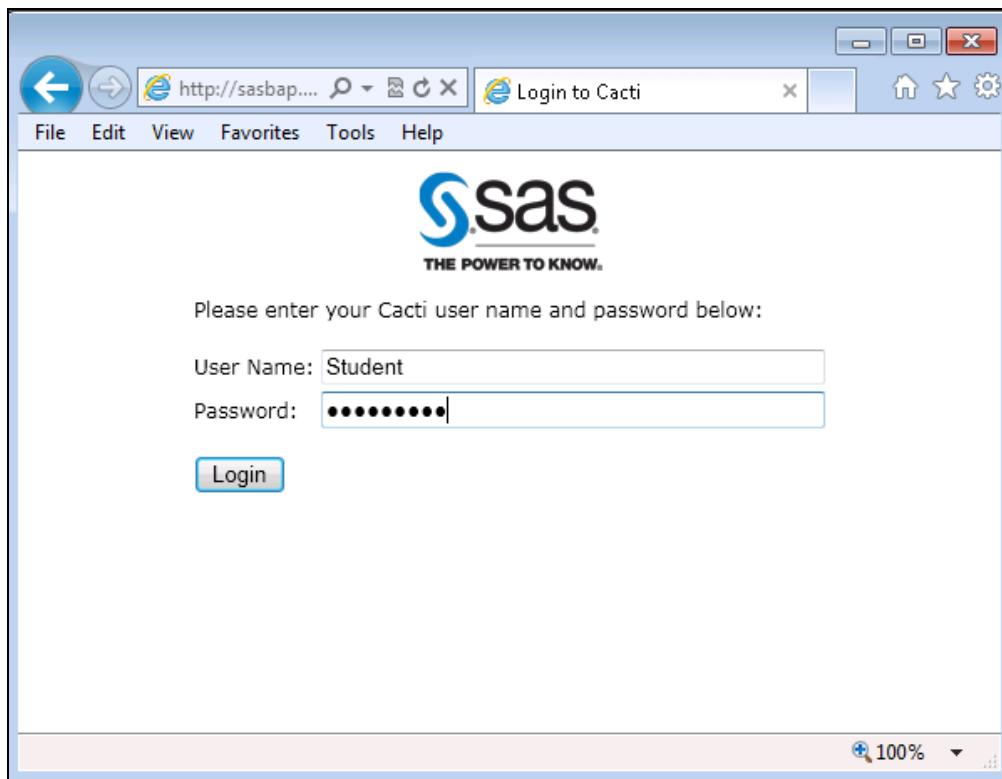
1. Log on to the remote desktop. Use the IP address provided by the instructor. The user ID is **Student** and the password is **Metadata0**.

Because this is the first time you are using your Student environment, you need to run a script to start the required SAS Grid Services. This is due to the unique configuration of the Student SAS Grid Manager environments. In a production environment, the SAS Grid Services start automatically when the grid servers are booted.

On the desktop of your student environment, double-click the **Start Grid Services** icon. A command window opens to run the script that starts the services.



2. Open Platform RTM for SAS by double-clicking the **Internet Explorer** icon. The home page is set to the Platform RTM for SAS web application's login page. The first time you connect to Platform RTM for SAS takes a little longer, so please give it time to come up.
3. Log on with the user ID **Student** and the password **Metadata0**.



4. Use the Run SAS Programs shortcut to start a set of SAS programs running on the grid.



- a. Double-click the **Run SAS Programs** shortcut icon on the desktop ( ). This calls a script that runs several SAS programs on the grid under four different users.
5. Explore the grid tab.
- a. Platform RTM for SAS takes the Student user to the grid tab. Notice that the grid tab is selected and as a result is red. By default the job information by host is displayed.

Actions	Host	Cluster	Type	Model	Load/Batch	CPU Fact	CPU Pct	RunQ 1m	Mem Usage	Page Rate	Page Max	Max Slots	Num Slots	Run Slots	SSUSP Slots	USUSP Slots	Reserve Slots
	sasbap	sas_cluster	NTX64	Intel_EM64T	Ok:Ok	60	65.41%	3.24	17.03%	21.18%	224	5	0	0	0	0	0

- b. Click **By Host** under Job Info. Host information is displayed.

Actions	Host	Cluster	Type	Model	Load/Batch	CPU Fact	CPU Pct	RunQ 1m	Mem Usage	Page Rate	Page Max	Max Slots	Num Slots	Run Slots	SSUSP Slots	USUSP Slots	Reserve Slots
	sasbap	sas_cluster	NTX64	Intel_EM64T	Ok:Ok	60	65.79%	2.45	24.18%	19.98%	189	5	2	2	0	0	0

- c. Click the host **sasbap** to drill down to the jobs running on it. The student environment has only one server (known in Platform RTM for SAS as a host), so all of the jobs are running on it. You will see a set of SAS programs running similar to the following:

JobID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time	Pend	Run
1118	Job_sleep3m_1	RUNNING	2	\student	-	-	07-18 11:21:16	0.1m	0.8m
1119	Job_sleep3m_1	RUNNING	2	\corey	-	-	07-18 11:21:21	0.1m	0.7m
1120	Job_sleep3m_1	RUNNING	2	\bryan	-	-	07-18 11:21:26	0.2m	0.6m
1121	Job_sleep3m_1	RUNNING	2	\aaron	-	-	07-18 11:21:31	0.3m	0.5m
1122	Job_sleep3m_2	RUNNING	2	\student	-	-	07-18 11:21:36	0.4m	0.4m

- d. Click a blue JobID for the Student user to see the details for that job.  
e. Click **By Queue** under Job Info. Queue information is displayed.

Actions	Queue Name	Cluster Name	Priority	Status Reason	Max Slots	Num Slots**	Run Slots	Pend Slots	Suspend Slots	AVG Pend
	normal	sas_cluster	30	Open:Active	5	2	2	0	0	-
	priority	sas_cluster	43	Open:Active	5	0	0	0	0	-
	idle	sas_cluster	20	Open:Active	5	0	0	0	0	-
	night	sas_cluster	40	Open:Inactive	5	0	0	0	0	-
	chkpnt_rerun_queue	sas_cluster	40	Open:Active	5	0	0	0	0	-
	short	sas_cluster	35	Open:Active	5	0	0	0	0	-
	license	sas_cluster	33	Open:Active	5	0	0	0	0	-
	owners	sas_cluster	43	Open:Active	5	0	0	0	0	-
	unicodecmd	sas_cluster	30	Open:Active	5	0	0	0	0	-

These are the default set of queues that are installed. In your SAS Grid Manager, you most likely have a different set of queues.

- f. Click **Details** under Job Info. Reset the status to **Active** and minimize Platform RTM for SAS.

**End of Demonstration**



## Exercises

---

### 1. Using Platform RTM

If you did not follow along with the demonstration, repeat the steps of the demonstration.

**End of Exercises**

## 2.2 SAS Management Console

---

### Objectives

- Describe SAS metadata.
- Describe SAS Management Console.
- Describe SAS Management Console with the SAS Grid Manager plug-in.

16

### SAS Metadata

SAS metadata is structured data that describes the location, structure, and settings for components on the platform for SAS Business Analytics.

- stored in a repository
- accessed via the SAS Metadata Server

SAS Management Console is the SAS client application to maintain SAS metadata.

17

In a SAS Grid Manager environment, SAS Management Console has the SAS Grid Manager plug-in that enables you to view your jobs, information about the hosts (servers), and queues in your SAS Grid Manager Environment.

In addition, you can perform some management operations for your SAS Grid Manager environment if the user has authorization.

## SAS Metadata

SAS Management Console contains plug-ins for managing the different areas of SAS metadata, for maintaining your SAS configuration, and for monitoring your SAS environment.

Some of the most popular plug-ins are the following:

- Grid Manager
- Metadata Manager
- Schedule Manager
- Server Manager
- User Manager

18

## Monitoring SAS Grid Execution

In a SAS Grid Manager environment, SAS Management Console with the SAS Grid Manager plug-in has the following features:

- monitoring features of jobs, hosts, and queues, with graphs
- some grid management features, but no configuration tools

19

## SAS Grid Manager Plug-in – Options

The screenshot shows the SAS Management Console interface. On the left, there is a tree view of management components under the 'Foundation' repository. A red arrow points from the 'Options...' button in the context menu of the 'Grid Monitoring Server - sasbap' node to a separate window titled 'Grid Server Options'. This options window contains two dropdown menus: 'Auto-Refresh Interval' set to '15 second(s)' and 'Time That Auto-Refresh is Active' set to '30 minute(s)'. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

20

Users can set options for the Grid Monitoring Server to indicate how often it refreshes the information that it displays.

## SAS Grid Manager Plug-in – Job Information

The screenshot shows the SAS Management Console interface. On the left, the tree view shows the 'Grid Monitoring Server - sasbap' node expanded, with 'Job Information' selected. A red arrow points from this selection to a large table on the right. The table has columns: Job ID, Job Name, User Name, and Status. It lists several entries, all of which are 'sleep 30' jobs run by '\sas'. The status column shows 'Running', 'Pending', 'Pending', 'Pending', 'Pending', 'Done', 'Done', 'Done', and 'Done'. Below this table is another table with columns: Submit Time, Start Time, End Time, and Execute Hosts. It lists multiple entries for May 13, 2014, at various times, all associated with the '\sas' user and the 'sasbap' host.

21

Under the SAS Grid Manager is the Grid Monitoring Server that shows three views of grid execution: job, host, and queue. Users can click the job, host, or queue to see information from the grid regarding the item that they selected. If users have permission, they can terminate, suspend, or resume jobs by right-clicking one of their jobs.

## SAS Grid Manager Plug-in – Job Information

The screenshot shows the SAS Management Console with the 'Job Information' section of the Grid Manager plug-in selected. The sidebar on the left lists 'Authorization Manager', 'Data Library Manager', and 'Grid Manager'. Under 'Grid Manager', 'Grid Monitoring Server - sasbap' is expanded, showing 'Job Information', 'Host Information', and 'Queue Information'. The main area displays a grid of jobs with the following data:

Job ID	Job Name	User	Status	Submit Time
672	LongRun	\Corey	Success	:10 PM Jun 20, 2014
671	LongRun	\stu	Success	:26 PM Jun 20, 2014

A context menu is open over the second row, listing options like 'Hold Column to Left', 'Sort Ascending', and 'Create Graph By Host'.

Users can see more columns, sort the columns, and create graphs by right-clicking the listed jobs.

22

## SAS Grid Manager Plug-in – Host Information

The screenshot shows the SAS Management Console with the 'Host Information' section of the Grid Manager plug-in selected. The sidebar on the left lists 'Authorization Manager', 'Data Library Manager', and 'Grid Manager'. Under 'Grid Manager', 'Grid Monitoring Server - sasbap' is expanded, showing 'Job Information', 'Host Information' (which is highlighted with a red box), and 'Queue Information'. The main area displays a grid of hosts with the following data:

Host Name	Host Model	Host Status	Host Type	Number of Jobs
sasbap	Intel_EM64T	ok	NTX64	0

23

Clicking the **Host Information** object displays the list of hosts (servers) in your grid with their status and other information about the host. If the user has the privilege, he can close or open hosts in the grid by right-clicking the host.

Right-click the heading row in the list of hosts in your grid to do the following:

- access menu items for sorting the rows
- choose additional columns to display
- change the user ID for retrieving the data from the grid

**SAS Grid Manager Plug-in – Queue Information**

Queue Name	Status	Total Jobs	Pending Jobs	Running Jobs
owners	Active	0	0	0
priority	Active	0	0	0
right	Inactive (Window)	0	0	0
chkpt_rerun_queue	Active	0	0	0
short	Active	0	0	0
license	Active	0	0	0
normal	Active	0	0	0
unicodecmd	Active	0	0	0
idle	Active	3	3	0

24

The set of queues in your grid is listed when you click the **Queue Information** object. If the user has the privilege, he can open, close, activate, or deactivate a queue by right-clicking a selected queue.

Right-click the heading row in the list of hosts in your grid to do the following:

- access menu items for sorting the rows
- choose additional columns to display
- change the user ID for retrieving the data from the grid.

## 2.02 Multiple Answer Poll

Which of the following does the SAS Grid Manager plug-in for SAS Management Console enable you to do?

- a. view the list of queues in your grid
- b. set the auto-refresh rate for the displayed information
- c. view the list of jobs in your grid
- d. view the list of hosts in your grid
- e. open and close hosts in your grid

25



## Exercises

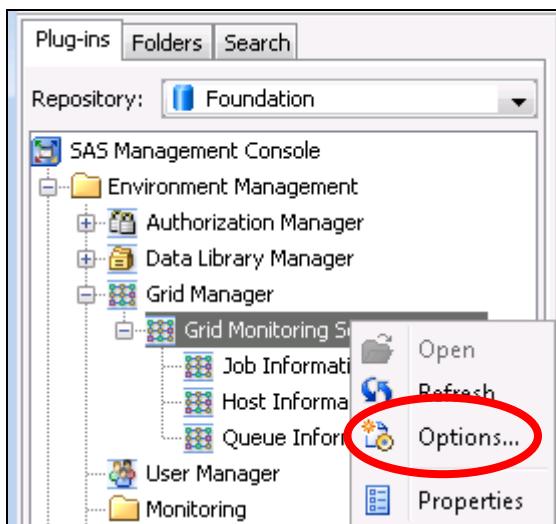
---

### 2. Monitoring Grid Execution Using SAS Management Console

Use the Run SAS Programs shortcut to start a set of SAS programs running on the grid.



- a. Double-click the **Run SAS Programs** shortcut icon on the desktop (). This calls a script that runs several SAS programs on the grid under four different users.
- b. Open SAS Management Console by double-clicking the desktop icon. Click **OK** in the Connection Profile prompt to use the My Server connection profile. Log on with the ID **Student** and the password **Metadata0**. This is the same user that you are logged in to as on your Windows machine.
- c. Expand the **Grid Manager** plug-in, and then expand **Grid Monitoring Server - sasbap**.
- d. Right-click **Grid Monitoring** and then choose **Options**.

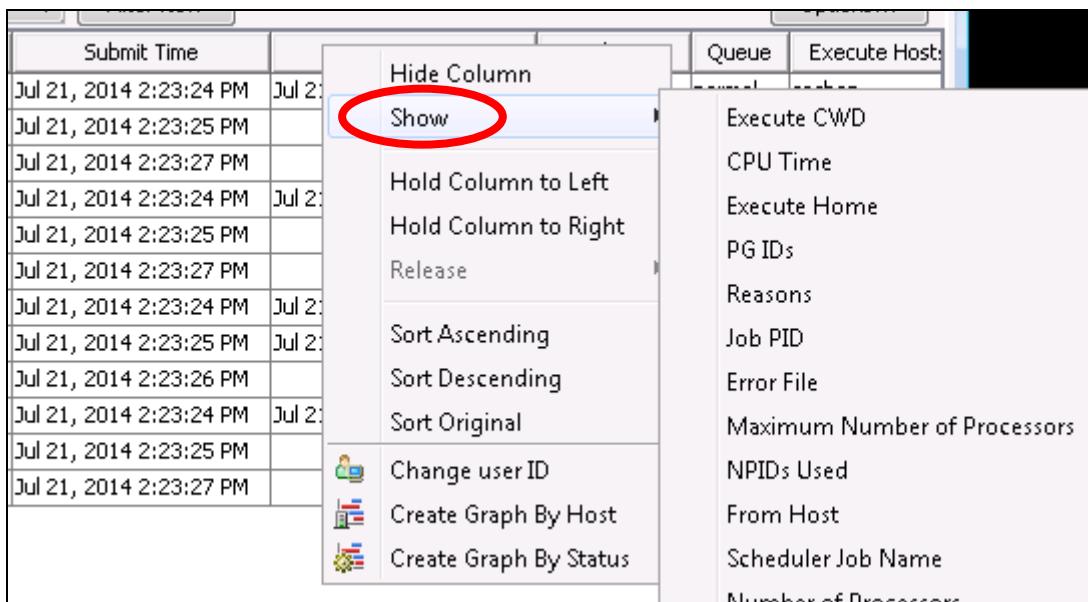


- 1) Set the **Auto-Refresh** interval to 15 seconds.
- 2) Set the **Time That Auto-Refresh is Active** to 5 hours.

This sets the Grid Manager plug-in so that it will automatically refresh the information that it is displaying every 15 seconds and continue do so for the next five hours.

- e. Click **Job Information**.
- f. Right-click one of the column names in the pane showing the job information and click

Show ⇔ **Memory Used** to add this column to the job information pane. Notice the list of columns and the information that you can display about each job.



- g. Record the number of grid jobs, hosts, and queues.
  - 1) Jobs \_\_\_\_\_
  - 2) Hosts \_\_\_\_\_
  - 3) Queues \_\_\_\_\_
- h. Click **Job Information** and then right-click the **Status** column heading. Choose to create a graph by status. Close the graph after you review it.
- i. Minimize SAS Management Console.

**End of Exercises**

## 2.3 Command Line Tools

---

### Objectives

- Describe command-line grid-monitoring tools.
- Use command-line grid-monitoring tools.

29

### SAS Grid Manager Command-Line Tools

Command-line tools are available to use if you use one of the following to submit jobs to the grid:

- SAS Foundation
- SAS Grid Manager Client Utility (SASGSUB)

30

If SAS Foundation is installed on your workstation and you use it to submit SAS programs to the grid, or if you have the SAS Grid Manager Client Utility (also known as SASGSUB), then you have the ability to run commands from a command prompt to monitor and view information about your grid.

## SAS Grid Manager Command-Line Tools

The most popular command-line tools that are used to monitor your grid and view information about your grid are the following:

- **bjobs**
- **bhosts**
- **bkill**
- **lsload**
- **bhist** or **bhist -l <job ID>**
- **bqueues**

31

Command	Description
<b>bjobs</b>	Lists the jobs in your grid.
<b>bhosts</b>	Lists the hosts in your grid along with the number of jobs.
<b>bkill</b>	Terminates jobs.
<b>lsload</b>	Displays the load information for each host in your grid.
<b>bhist</b>	Displays the history of the job in your grid.
<b>bqueues</b>	Displays the queues in the grid.

## Viewing Your Grid Jobs

To view information about jobs in your grid, use the command **bjobs**.

- **bjobs** lists all active jobs for user.
- **bjobs <job ID>** lists a job for the given job ID.
- **bjobs -a** lists all jobs for all statuses.
- **bjobs -l <job ID>** lists the details for the job ID.
- **bjobs -?** opens the Help for this command.

32

## Stopping Your Jobs

To stop jobs that are running in your grid, use the command **bkill**.

- **bkill <job ID>** kills (stop immediately) the job specified by job ID.
- **bkill 0** kills all of the user's active jobs.

33

## Viewing the Load of Your Hosts

To view how busy the hosts are in your grid, use the command **lload**.

```
[sas@sasserver sasinside]# lload
```

HOST_NAME	status	r15s	r1m	r15m	ut	pg	ls	it	tmp	swp	mem
sasgridn02.demo	ok	0.0	0.0	0.0	0%	0.0	0	16608	254G	21G	15G
sasgridn01.demo	ok	0.0	0.0	0.0	0%	0.0	1	1924	255G	21G	15G
sasserver.demo.	ok	0.4	0.1	0.1	5%	0.0	1	5	209G	21G	9470M



## Exercises

---

### 3. Monitoring Grid Execution Using Command Line Tools

- a. Open a command window by double-clicking the desktop shortcut for the Command Prompt.
- b. Determine whether you have any jobs running on the grid.
  - 1) Open a command window using this shortcut on the desktop: **Command Prompt**
  - 2) Type **bjobs -u all**. Press Enter. You should get this response: **No unfinished jobs found**
- c. Use the **Submit\_SAS\_Programs.cmd** script to run several jobs on the grid as different users, including the user that you are logged on as (**Student**).
  - 1) Change to the directory containing the script. Use the following command:  
**cd C:\grid\Grid Demo Docs\Monitoring Grid**
  - 2) Run the script by entering **Submit\_SAS\_Programs.cmd**. Press Enter.
  - 3) View all running jobs. Enter **bjobs -u all**. Press Enter.
  - 4) How many jobs are running? \_\_\_\_\_
  - 5) What are the job IDs for the running jobs? \_\_\_\_\_
  - 6) Are there any pending jobs? If, so how many? \_\_\_\_\_
  - 7) Determine how many jobs are running for the user you are logged on as (**Student**). Enter the following command in the command window: **bjobs**. Press Enter.
  - 8) How many jobs is your user attempting to run? \_\_\_\_\_
  - 9) View the detailed information for one of your jobs that is pending by entering the following:  
**bjobs -l <job ID>** for one of the jobs listed with a status of PEND  
10) What is listed as the reason for your job being in a pending state? \_\_\_\_\_
- d. Terminate one of your jobs. Enter **bkill <job ID>**. Press Enter.  
Your jobs will be any job running as the Student user.
- e. View the load of the grid by entering **lload**. Press Enter.
- f. View the hosts in the grid by entering **bhosts**. Press Enter.

**End of Exercises**

## 2.4 Solutions

---

### Solutions to Exercises

#### 1. Using Platform RTM

- Log on to the remote desktop. Use the IP address provided by the instructor. The user ID is Student and the password is Metadata0.

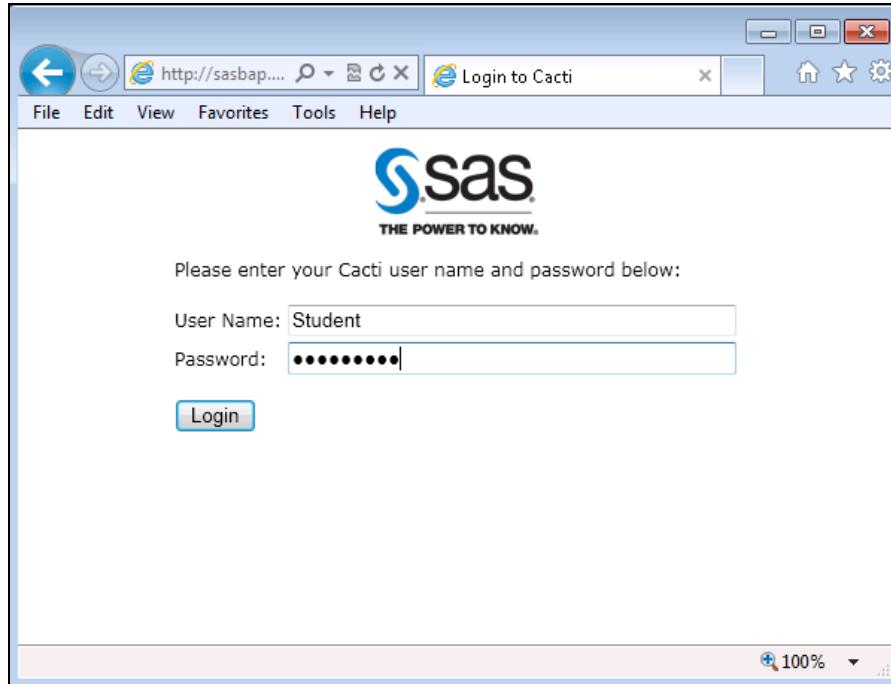
Because this is the first time using your Student environment, you need to run a script to start the required services. This is due only to configuration and the internal environment that runs the student virtual machine environments.

On the desktop of your student environment, double-click the **Start Grid Services** icon



(**Start Grid Services**). A command window opens to run the script that starts the services.

- Open Platform RTM for SAS by double-clicking the **Internet Explorer** icon. The home page is set to the Platform RTM for SAS web application's login page. The first time you connect to Platform RTM for SAS takes a little longer, so please give it time to come up.
- Log on with the user ID **Student** and the password **Metadata0**.



- Use the Run SAS Programs shortcut to start a set of SAS programs running on the grid.



Double-click the **Run SAS Programs** shortcut icon on the desktop ([Run SAS programs](#)).

This calls a script that runs several SAS programs on the grid under four different users.

**e. Explore the grid tab.**

Platform RTM for SAS takes the Student user to the grid tab. Notice that the grid tab is selected and as a result is red. By default the job information by host is displayed.

Actions	Host Name	Cluster	Type	Model	Load/Batch	CPU Fact Pct	CPU 1m Usage	RunQ Mem Usage	Page Rate	Page Max Slots	Max Num Slots	Run SSUSP Slots	SSUSP Slots	Reserve Slots	
	sasbap	sas_cluster	NTX64	Intel_EM64T	Ok:Ok	60	65.41%	3.24	17.03%	21.18%	224	5	0	0	0

**f. Click **By Host** under Job Info. Host information is displayed.**

Actions	Host Name	Cluster	Type	Model	Load/Batch	CPU Fact Pct	CPU 1m Usage	RunQ Mem Usage	Page Rate	Page Max Slots	Max Num Slots	Run SSUSP Slots	SSUSP Slots	Reserve Slots	
	sasbap	sas_cluster	NTX64	Intel_EM64T	Ok:Ok	60	65.79%	2.45	24.18%	19.98%	189	5	2	2	0

- g. Click the host **sasbap** to drill down to the jobs running on it. The student environment has only one server (known in Platform RTM for SAS as a host), so all of the jobs are running on it. You will see a set of SAS programs running similar to the following:

The screenshot shows the 'Batch Job Filters' interface. The left sidebar lists categories like Job Info, By Host, By Host Group, By Queue, Details, User/Group Info, Users, Groups, Load Info, Host, Host Group, Host Info, Servers, and Groups. The 'Host' category is selected. The main area displays a table of jobs with columns: JobID, Name, Status, State Changes, User ID, CPU Usage, CPU Effic, Start Time, Pend, and Run. The table shows five entries, all labeled 'Job\_sleep3m\_1' and 'RUNNING'. The bottom navigation bar includes buttons for Warning Efficiency, Alarm Efficiency, Flapping, Dependencies, Invalid Dependencies, Exited, and Exclusive.

JobID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time	Pend	Run
1118	Job_sleep3m_1	RUNNING	2	\student	-	-	07-18 11:21:16	0.1m	0.8m
1119	Job_sleep3m_1	RUNNING	2	\corey	-	-	07-18 11:21:21	0.1m	0.7m
1120	Job_sleep3m_1	RUNNING	2	\bryan	-	-	07-18 11:21:26	0.2m	0.6m
1121	Job_sleep3m_1	RUNNING	2	\aaron	-	-	07-18 11:21:31	0.3m	0.5m
1122	Job_sleep3m_2	RUNNING	2	\student	-	-	07-18 11:21:36	0.4m	0.4m

- h. Click a blue JobID for the Student user to see the details for that job.  
i. Click **By Queue** under Job Info. Queue information is displayed.

The screenshot shows the 'Batch Queue Filters' interface. The left sidebar lists categories like Job Info, By Host, By Host Group, **By Queue**, Details, User/Group Info, Users, Groups, Load Info, Host, Host Group, Host Info, Servers, and Groups. The 'By Queue' category is selected and highlighted with a red oval. The main area displays a table of queues with columns: Actions, Queue Name, Cluster Name, Priority, Status Reason, Max Slots, Num Slots\*\*, Run Slots, Pend Slots, Suspend Slots, and AVG Pend. The table shows nine entries, each with a magnifying glass icon and a color-coded status. The bottom navigation bar includes a 'Show Previous' button and a 'Showing Rows 1 to 9 of 9 [1]' message.

Actions	Queue Name	Cluster Name	Priority	Status Reason	Max Slots	Num Slots**	Run Slots	Pend Slots	Suspend Slots	AVG Pend
	normal	sas_cluster	30	Open:Active	5	2	2	0	0	-
	priority	sas_cluster	43	Open:Active	5	0	0	0	0	-
	idle	sas_cluster	20	Open:Active	5	0	0	0	0	-
	night	sas_cluster	40	Open:Inactive	5	0	0	0	0	-
	chkptn_rerun_queue	sas_cluster	40	Open:Active	5	0	0	0	0	-
	short	sas_cluster	35	Open:Active	5	0	0	0	0	-
	license	sas_cluster	33	Open:Active	5	0	0	0	0	-
	owners	sas_cluster	43	Open:Active	5	0	0	0	0	-
	unicodecmd	sas_cluster	30	Open:Active	5	0	0	0	0	-

- j. These are the default set of queues that are installed. In your SAS Grid Manager, you most likely have a different set of queues.  
k. Click **Details** under Job Info. Reset the status to **Active** and minimize Platform RTM for SAS.

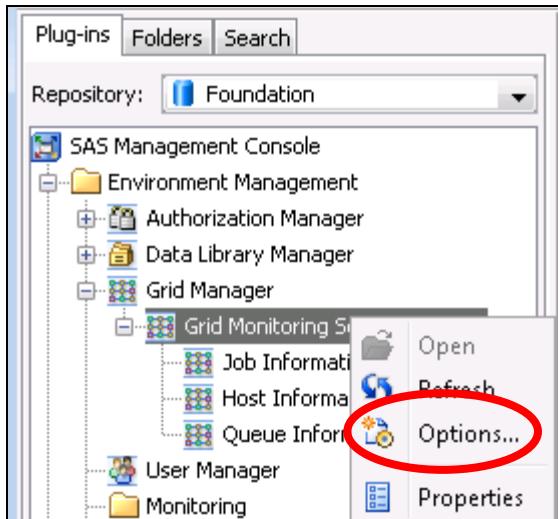
## 2. Monitoring Grid Execution Using SAS Management Console

Use the Run SAS Programs shortcut to start a set of SAS programs running on the grid.



- a. Double-click the **Run SAS Programs** shortcut icon on the desktop. This calls a script that runs several SAS programs on the grid under four different users.  
b. Open SAS Management Console by double-clicking the desktop icon. Click **OK** in the Connection Profile prompt to use the My Server connection profile. Log on with the ID **Student** and the password **Metadata0**. This is the same user that you are logged in to as on your Windows machine.

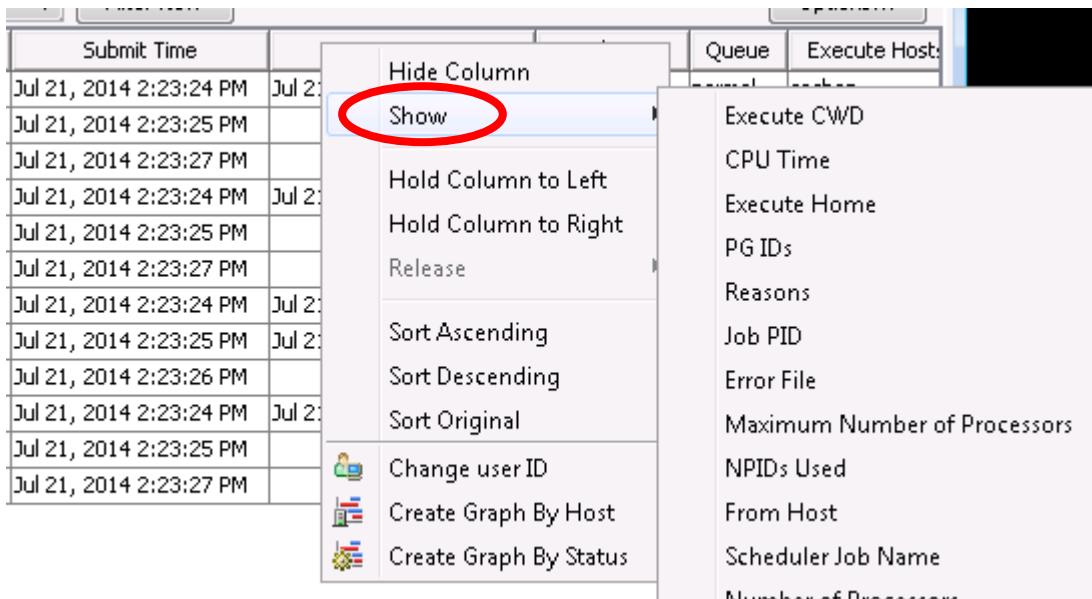
- c. Expand the **Grid Manager** plug-in, and then expand **Grid Monitoring Server – sasbap**.
- d. Right-click **Grid Monitoring** and then select **Options**.



- 1) Set the **Auto-Refresh** interval to **15** seconds.
- 2) Set **Time That Auto-Refresh is Active** to **5** hours.

This sets the Grid Manager plug-in so that it will automatically refresh the information that it is displaying every 15 seconds and continue do so for the next 5 hours.

- e. Click **Job Information**.
- f. Right-click one of the column names in the pane showing the job information and select **Show**  $\Rightarrow$  **Memory Used** to add this column to the job information pane. Notice the list of columns and the information that you can display about each job.



- g. Record the number of grid jobs, hosts, and queues.

1) Jobs      **Running: 5, Pending 7**

2) Hosts      **1**

3) Queues      **9**

 Your list of jobs should be similar to the following:

Job ID	Job Name	User Name /	Status	Submit Time	Start Time	End Time	Queue	Execute Host
1106	Job_sleep3m_1	.\\Aaron	Running	Jul 21, 2014 2:23:24 PM	Jul 21, 2014 2:23:33 PM		normal	sasbap
1110	Job_sleep3m_2	.\\Aaron	Pending	Jul 21, 2014 2:23:25 PM			normal	
1116	Job_sleep3m_3	.\\Aaron	Pending	Jul 21, 2014 2:23:27 PM			normal	
1105	Job_sleep3m_1	.\\Bryan	Running	Jul 21, 2014 2:23:24 PM	Jul 21, 2014 2:23:28 PM		normal	sasbap
1111	Job_sleep3m_2	.\\Bryan	Pending	Jul 21, 2014 2:23:25 PM			normal	
1114	Job_sleep3m_3	.\\Bryan	Pending	Jul 21, 2014 2:23:27 PM			normal	
1107	Job_sleep3m_1	.\\Corey	Running	Jul 21, 2014 2:23:24 PM	Jul 21, 2014 2:23:38 PM		normal	sasbap
1109	Job_sleep3m_2	.\\Corey	Running	Jul 21, 2014 2:23:25 PM	Jul 21, 2014 2:23:48 PM		normal	sasbap
1113	Job_sleep3m_3	.\\Corey	Pending	Jul 21, 2014 2:23:26 PM			normal	
1108	Job_sleep3m_1	.\\student	Running	Jul 21, 2014 2:23:24 PM	Jul 21, 2014 2:23:43 PM		normal	sasbap
1112	Job_sleep3m_2	.\\student	Pending	Jul 21, 2014 2:23:25 PM			normal	
1115	Job_sleep3m_3	.\\student	Pending	Jul 21, 2014 2:23:27 PM			normal	

- h. Click **Job Information** and then right-click the **Status** column heading. Choose to create a graph by status. Close the graph after you review it.

- i. Minimize SAS Management Console.

### 3. Monitoring Grid Execution Using Command Line Tools

- a. Open a command window by double-clicking the desktop shortcut for the Command Prompt.

- b. Determine whether you have any jobs running on the grid.

1) Open a command window using the desktop shortcut **Command Prompt**.

2) Type **bjobs -u all**. Press Enter. You should get the response **No unfinished jobs found**.

- c. Use the **Submit\_SAS\_Programs.cmd** script to run several jobs on the grid as different users including the user that you are logged on as (**Student**).

1) Change to the directory containing the script. Use the following command:

**cd C:\\grid\\Grid Demo Docs\\Monitoring Grid**

2) Run the script by entering **Submit\_SAS\_Programs.cmd**. Press Enter.

a) Wait approximately one minute so that all of the jobs can start running.

3) View all running jobs. Enter **bjobs -u all**. Press Enter.

4) How many jobs are running? **5**

5) What are the job IDs for the running jobs? **Varies**

6) Are there any pending jobs? If, so how many? **10**

- 7) Determine how many jobs are running for the user you are logged on as (**Student**). Enter the following command in the command window: **bjobs**. Press Enter.
- 8) How many jobs is your user attempting to run? **3**
- 9) View the detailed information for one of your jobs that is pending by entering the following:  
**bjobs -l <job ID>** for one of the jobs listed with a status of PEND.

The screen capture below depicts output that you will see:

```
C:\Users\student>bjobs -l 1131
Job <1131>, Job Name <Job_sleep3m_3>, User <\student>, Project <default>, Status <PEND>, Queue <normal>, Command <C:\SAS\Config\Lev1\SAS App\GridServer\sasgrid "SASBATCHSASPGM:'sleep3.sas'" SASWANTJOBINFO:1 "SASBATCHWRKDIR:'\\sasbap\gridwork\student\SAS GSUB-2014-07-21_15.06.13.242_Job_sleep3m_3'">
Mon Jul 21 15:06:13: Submitted from host <sasbap> CWD <c:\grid\Grid Demo Docs\Monitoring Grid>, Requested Resources < SASApp >;
PENDING REASONS:
Job slot limit reached: 1 host;

SCHEDULING PARAMETERS:
      r15s    r1m    r15m    ut      pg      io      ls      it      tmp      swp      mem
loadSched   -      -      -      -      -      -      -      -      -      -      -
loadStop    -      -      -      -      -      -      -      -      -      -      -
```

- 10) What is listed as the reason for your job being in a pending state? **Job slot limit reached: 1 host;**
- d. Terminate one of your jobs. Enter **bkill <job ID>**. Press Enter.  
Your jobs will be any job running as the Student user.  
Verify that your job is no longer executing by entering **bjobs -u all**. Press Enter.  
The job you just killed should not be listed as running.
- e. View the load of the grid by entering **lsload**. Press Enter.
- f. View the hosts in the grid by entering **bhosts**. Press Enter.

**End of Solutions**

## Solutions to Student Activities

### 2.01 Multiple Answer Poll – Correct Answer

Which of the following does Platform RTM for SAS enable you to do?

- a. view job details
- b. view queue information
- c. subset job information based on status
- d. subset job information based on user

12

### 2.02 Multiple Answer Poll – Correct Answers

Which of the following does the SAS Grid Manager plug-in for SAS Management Console enable you to do?

- a. view the list of queues in your grid
- b. set the auto-refresh rate for the displayed information
- c. view the list of jobs in your grid
- d. view the list of hosts in your grid
- e. open and close hosts in your grid

26

# Chapter 3 Using SAS® Management Console and Schedule Manager with a SAS® Grid

<b>3.1 Scheduling on a SAS Grid.....</b>	<b>3-3</b>
Demonstration: Creating and Scheduling a Flow .....	3-14
Exercises .....	3-20
<b>3.2 Solutions .....</b>	<b>3-21</b>
Solutions to Exercises .....	3-21



## 3.1 Scheduling on a SAS Grid

### Objectives

- List the benefits of scheduling.
- Describe how SAS integrates with the scheduling functionality of the Platform Suite for SAS.
- Describe the components used for scheduling.
- Describe how SAS integrates with Process Manager to run scheduled SAS programs.

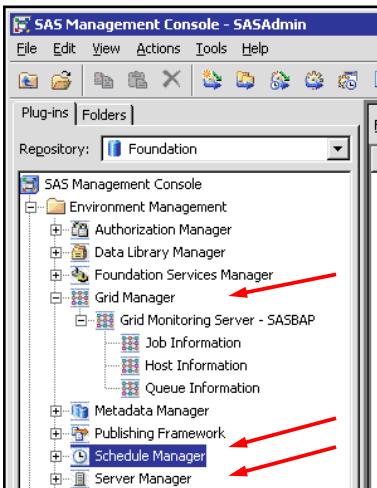
2

### Scheduling Key Benefits

- Define and maintain flows. Flows enable you to define
  - scheduling triggers
  - job dependencies, file dependencies, and time dependencies.
- Automate the timely execution of SAS jobs.
- Run jobs at appropriate times based on triggers and dependencies of other jobs.
- Leverage a SAS grid environment.

3

## SAS Management Console – Scheduling

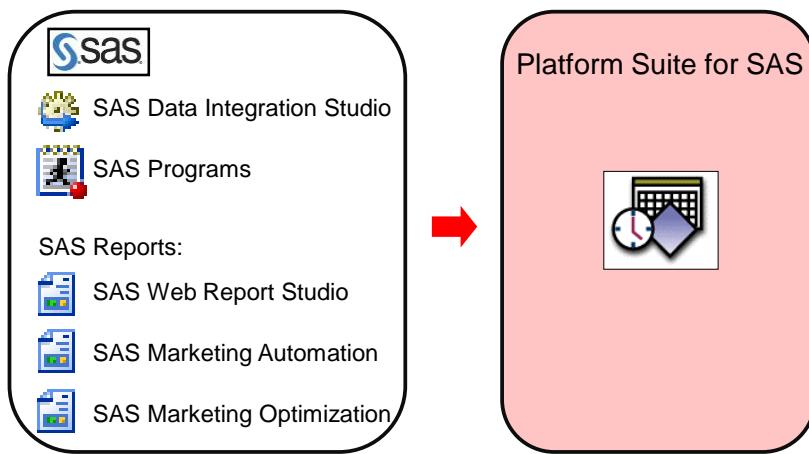


Scheduling in a grid environment uses the following:

- SAS Grid Manager is used to observe scheduled jobs running in the grid.
- SAS Server Manager is used to define, configure, and manage grid servers to support scheduling.
- SAS Schedule Manager is used to schedule execution of a SAS program.

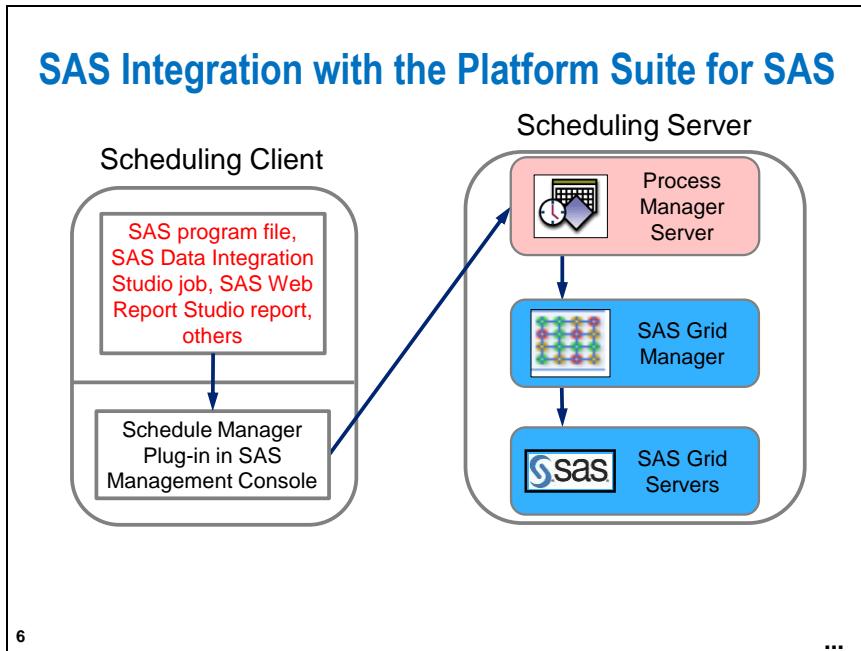
4

## Scheduling Jobs with the Platform Suite for SAS

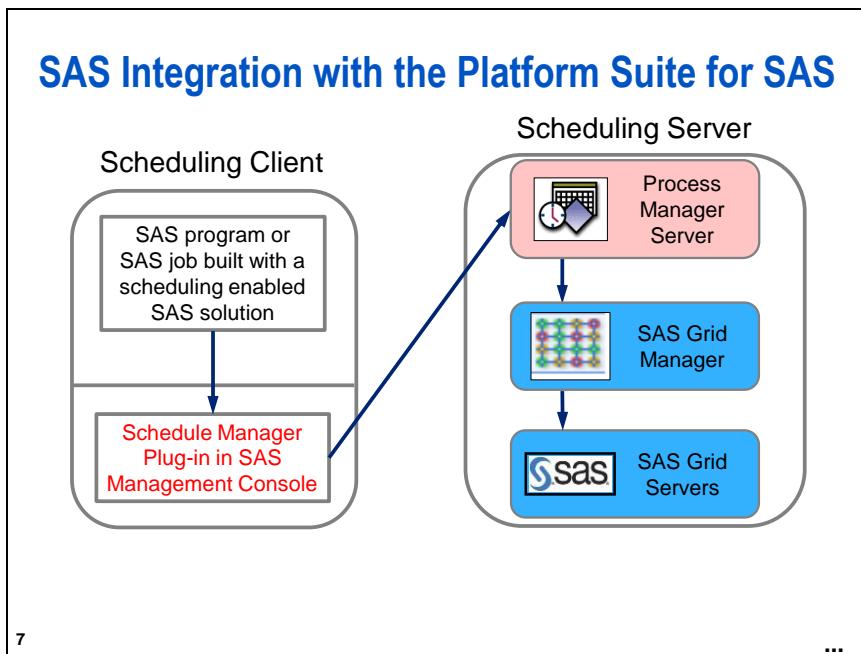


5

Job processes developed through SAS programs, SAS Data Integration Studio, SAS Web Report Studio, SAS Marketing Automation, and SAS Marketing Optimization can be scheduled using the components contained in the Platform Computing Inc. The Platform Suite for SAS is bundled with SAS Grid Manager. It supports simple or complex flows that can contain multiple jobs and rules that define when flows execute, the order in which jobs within flows execute, and other triggers and job dependencies. In conjunction with SAS Grid Manager, the Platform Suite for SAS enables you to distribute flows across a grid of compute hosts.

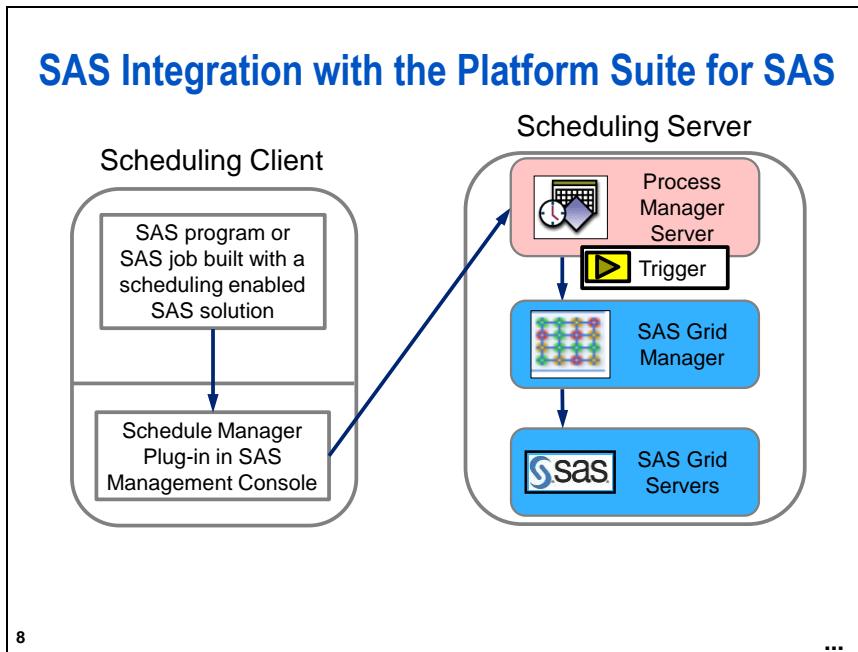


Tasks that can be scheduled include SAS programs, jobs built with SAS Data Integration Studio, SAS Web Report Studio reports, and processes built with SAS Marketing Automation and SAS Marketing Optimization. Each type of SAS process that is scheduled can be directed to execute on the specific type of required SAS batch server.

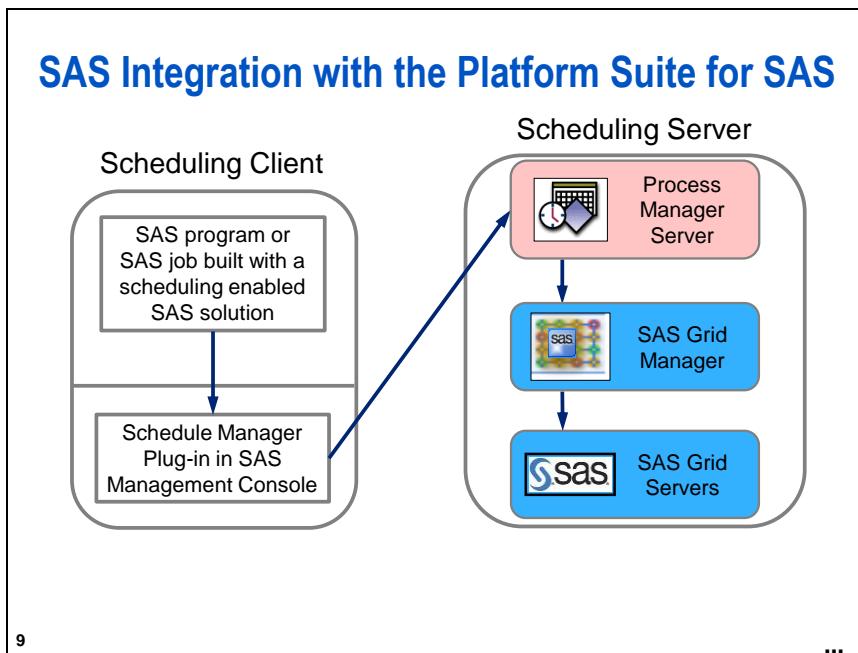


SAS Management Console is the primary SAS client scheduling interface for designing and submitting flows to the Platform Process Manager server.

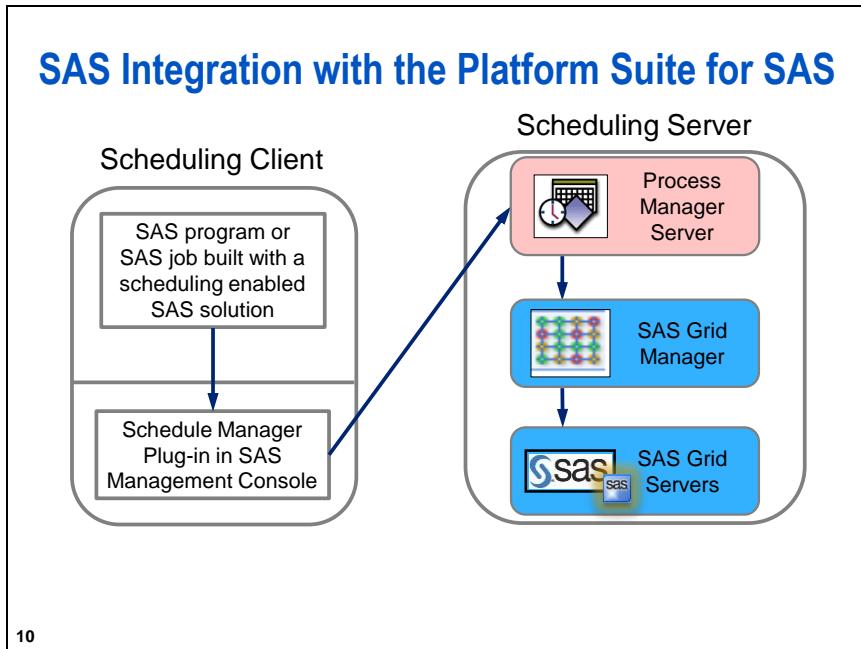
SAS Web Report Studio and SAS Marketing Automation also provide an interface for scheduling. After it is created, the scheduled flow is visible from the Schedule Manager plug-in within the SAS Management Console application.



Process Manager, which is an enterprise scheduling server, watches for the trigger to occur for the scheduled program. After the trigger occurs (time of day is met, a file arrives, and so on), Process Manager requests that the program is executed on the grid.



SAS Grid Manager receives the request to run the scheduled program and it then finds the least busy server to run the program.



Finally, the program runs on one of the grid servers.

### 3.01 Multiple Answer Poll

What types of SAS processes does your organization need to schedule?

- a. SAS programs
- b. SAS Data Integration Studio jobs
- c. SAS Web Report Studio reports
- d. SAS Marketing Automation processes
- e. other SAS processes

11

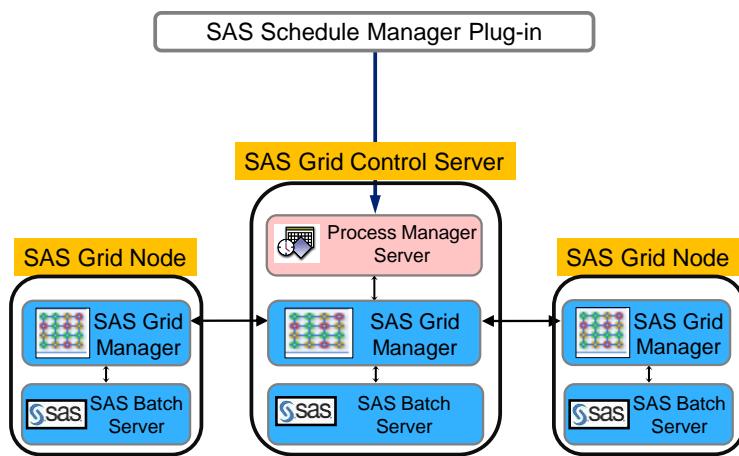
## Scheduling on a Grid

Creating new scheduled jobs is done in two ways:

- using features built into SAS applications or solutions
- manually using the Schedule Manager in SAS Management Console

12

## Scheduling on a Grid



13

Platform Suite for SAS is included with SAS Grid Manage. To execute scheduled flows on the grid, Process Manager server runs on the same host as the SAS Grid Control server. LSF and the SAS Batch Server are also installed and configured on each of the grid hosts that are made available to execute jobs in scheduled flows. The LSF master communicates with each of the LSF hosts to determine computing resource availability and directs jobs to the chosen host machines for execution.

The SAS Batch Server is actually a template used to create the entire command for running a batch SAS program using Platform Process Manager. Therefore, there is not a SAS Batch Server running and waiting to run the batch programs. Instead, during the scheduling process the information from the SAS Batch Server is used to build the command that Platform Process Manager uses to run the SAS Batch program. It includes environmental information such as the path for the log files and other settings.

## SAS Management Console: Scheduling a SAS DATA Step Program

The first step is to deploy a SAS program..

14

Scheduling is discussed in detail in the SAS® Enterprise Scheduling with Platform Suite for SAS® course.

## SAS Management Console: Scheduling a SAS DATA Step Program

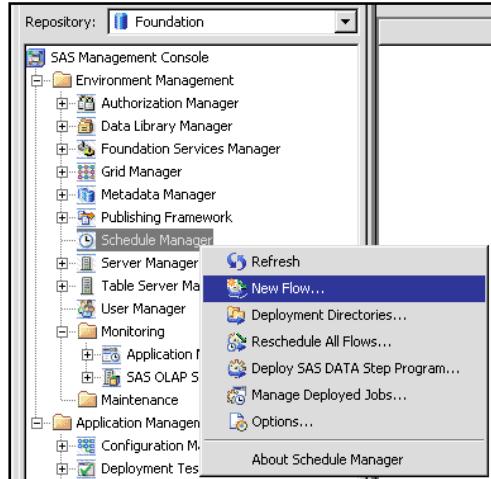
You can select a specific source program from a directory of jobs.

15

*continued...*

## SAS Management Console: Scheduling a SAS DATA Step Program

The next step is to define a flow.



16

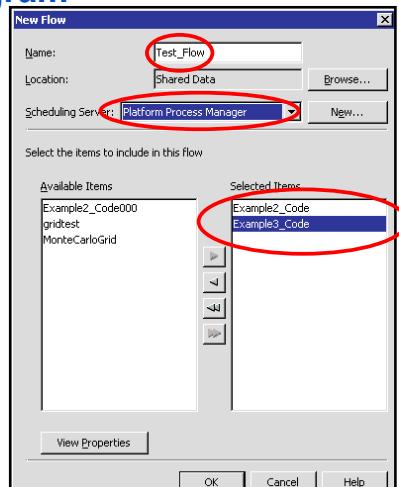
continued...



To view the pop-up menu, right-click the **Schedule Manager** plug-in.

## SAS Management Console: Scheduling a SAS DATA Step Program

- Name the flow.
- Select the server.
- Select the jobs.



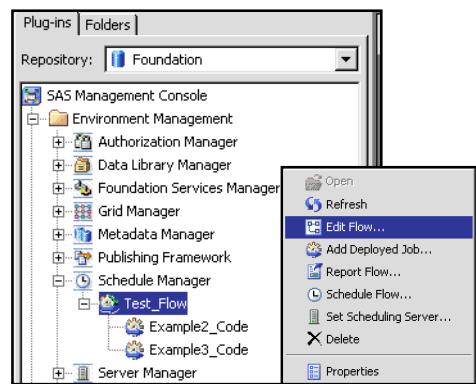
17

continued...

When you schedule to the grid, the server that is used is Platform Process Manager.

## SAS Management Console: Scheduling a SAS DATA Step Program

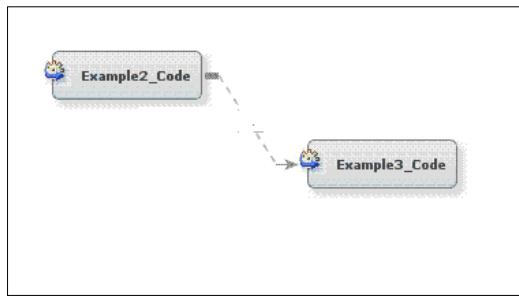
If you select **Edit Flow**, you can arrange the order of your programs within the flow, including extensive conditional logic.



18

## SAS Management Console: Scheduling a SAS DATA Step Program

**Edit Flow** displays a process flow diagram where you can control the order of jobs.

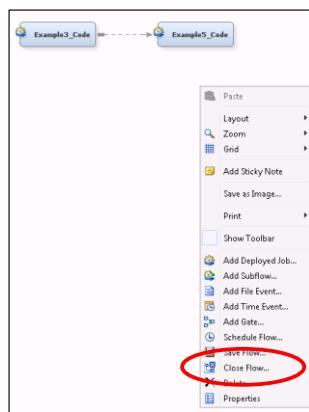


19

*continued...*

## SAS Management Console: Scheduling a SAS DATA Step Program

**Close and Save** the Flow to return to the main SAS Management Console windows.

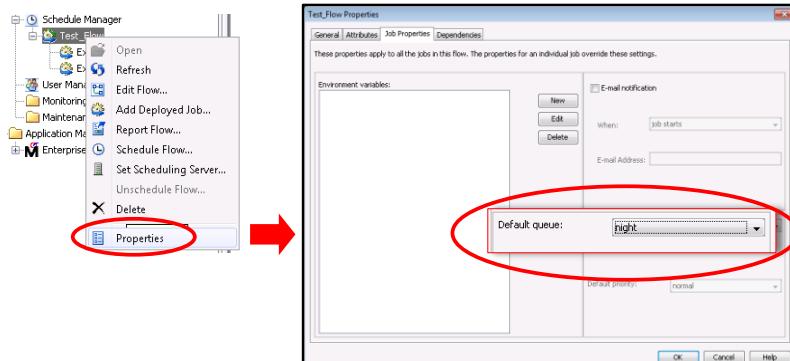


20

continued...

## SAS Management Console: Scheduling a SAS DATA Step Program

Set the Properties of the Flow, such as the queue for the scheduled job to be submitted to.



21

continued...

## SAS Management Console: Schedule Manager

The last step is to schedule the flow. The example here uses **Run Now**.

22



## Creating and Scheduling a Flow

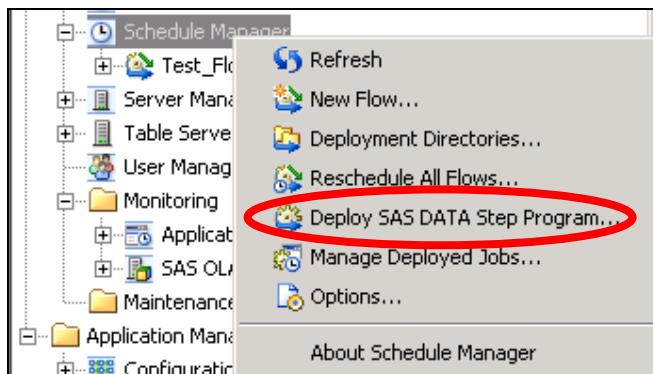
- If necessary, log on to SAS Management Console. Use the ID **Ahmed** and the password **Student1**.



- Expand the **Schedule Manager** plug-in.

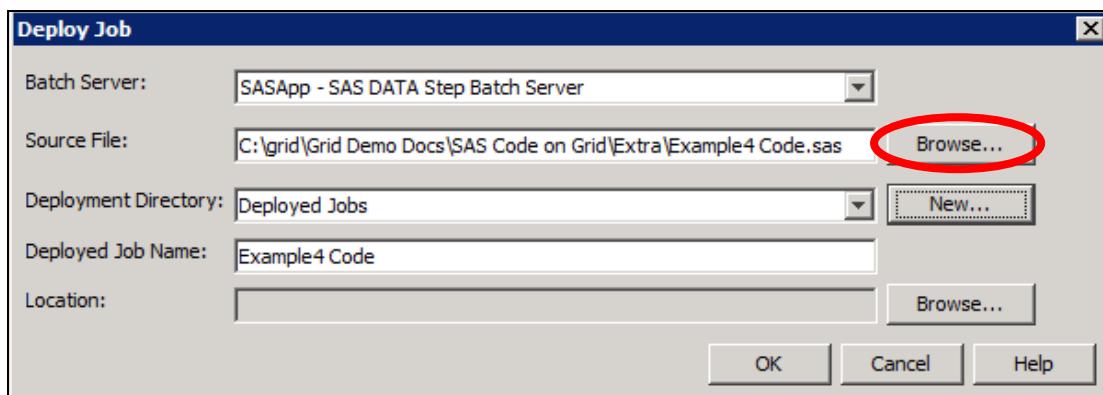
- Deploy the SAS program **Example4\_Code.sas**.

- Right-click **Schedule Manager** and select **Deploy SAS DATA Step Program**.

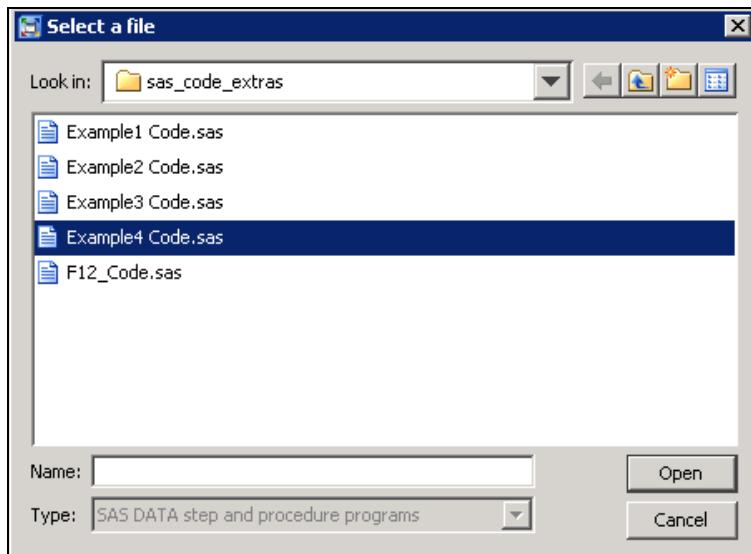


- Leave the batch server set to SASApp - SAS DATA Step Batch Server.

- Click **Browse** to select the source file.



- 4) Navigate to **C:\Grid\Grid Demo Docs\SAS Code on Grid\Extra**. Select the file **Example4 Code.sas**. Click **Open**.

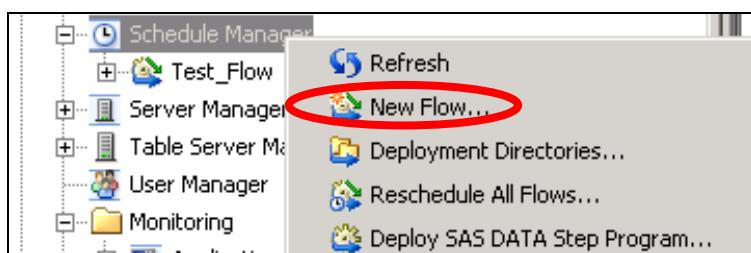


- 5) Keep the default values for the Deployment Directory and Deployed Job Name.
- 6) Click **OK**.
- 7) Click **OK** to convert the filename for the deployed code to a valid name.
- 8) When the message appears and indicates that the job was successfully deployed, click **OK**.

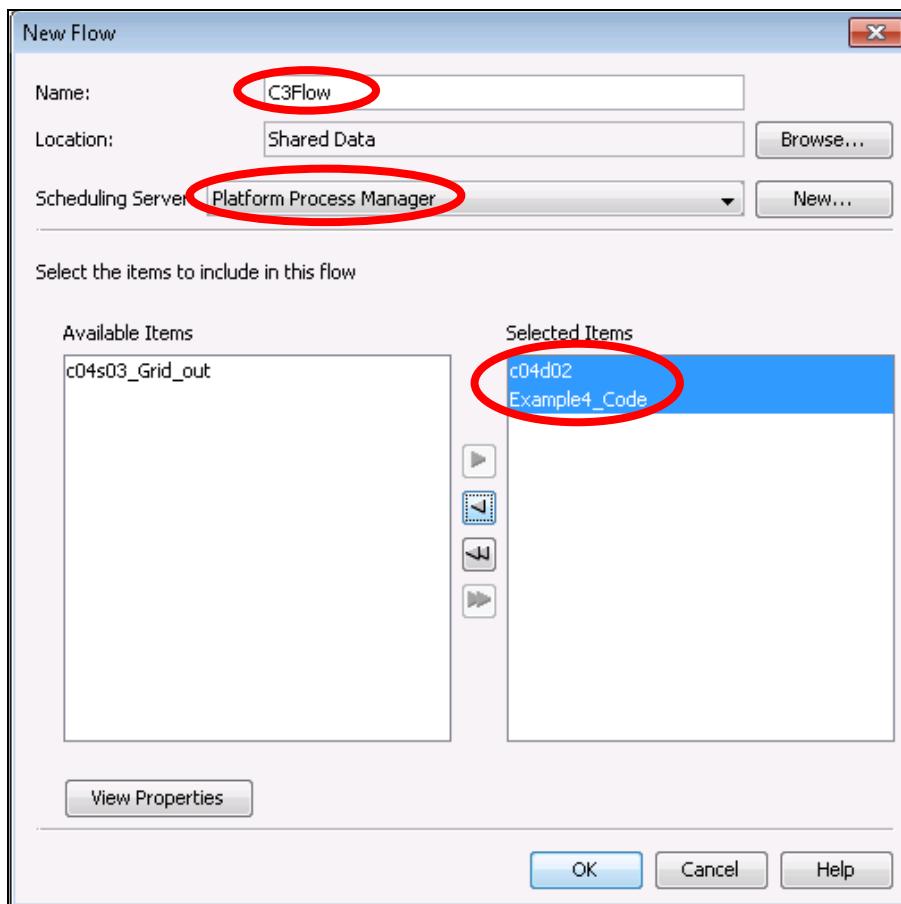


- b. Define a new flow involving two deployed programs.

- 1) Right-click **Schedule Manager** and select **New Flow**.

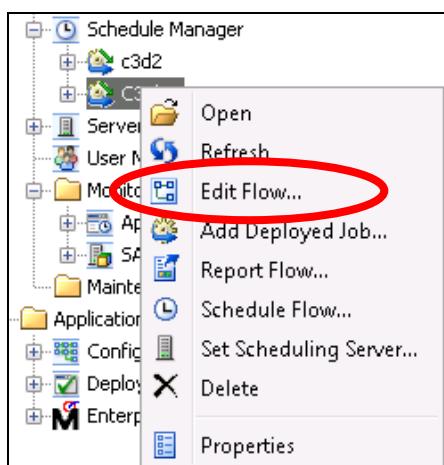


- 2) Name the new flow **C3\_Flow**. Select **Platform Process Manager** as the Scheduling Server. Move **c04d02** and **Example4\_Code** to the Selected Items list box. Click **OK**.

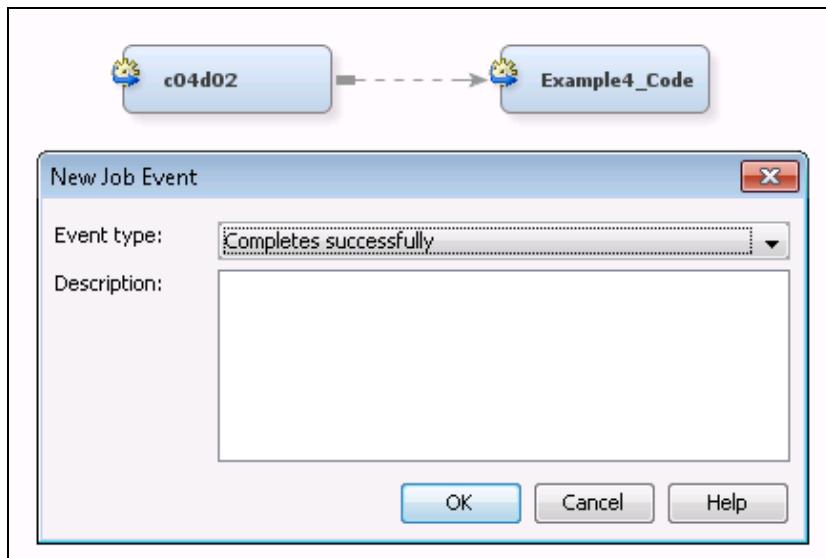


c. Edit the flow.

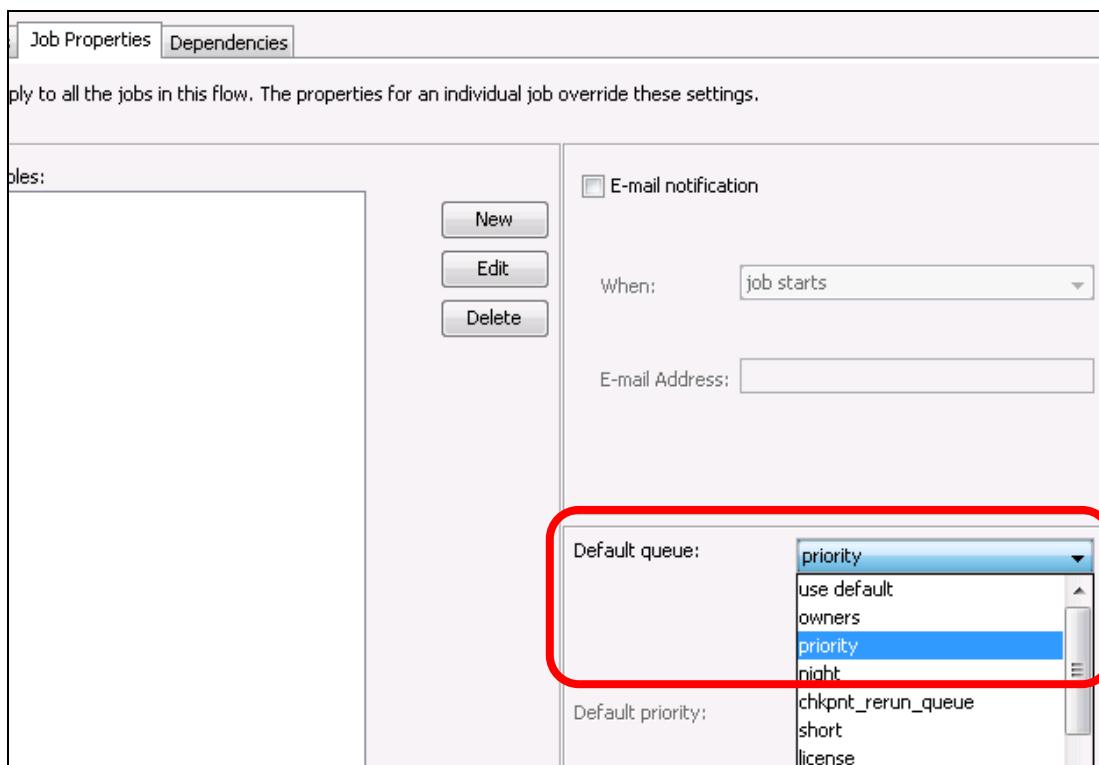
- 1) Expand **Schedule Manager**. Right-click **C3\_Flow** and select **Edit Flow...**



- 2) In the process flow diagram, draw a line that connects the two Code nodes. In the dialog box, click **OK**.

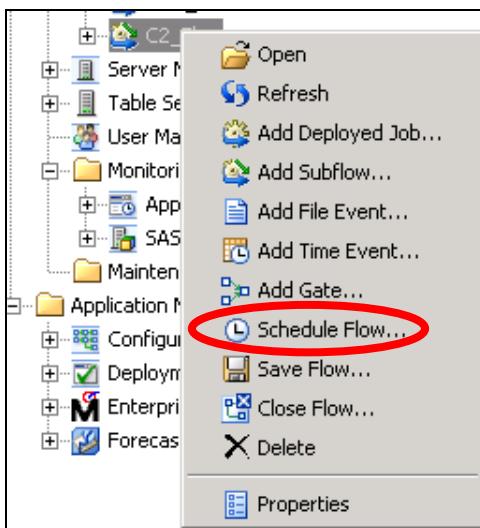


- 3) Right-click in the white space of your flow and select **Close Flow**. Click **Yes** to save changes.
- d. Edit the **properties** of the **C3\_Flow** to set the queue for the flow to the priority queue.
- 1) Right-click **C3\_Flow** and click **Properties**
  - 2) Click the **Job Properties** tab and set the Default queue to the **priority** queue.

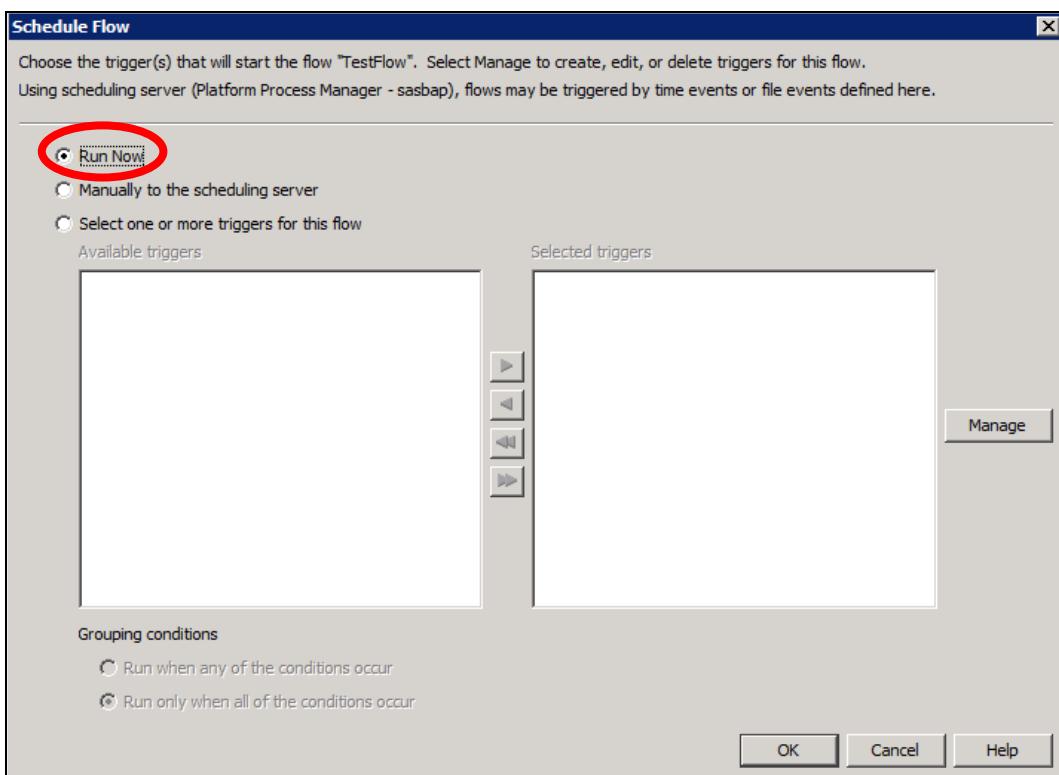


e. Schedule your new flow for immediate execution.

- 1) Right-click **C3\_Flow** and select **Schedule Flow**. If prompted, log on as **\Ahmed** with a password of **Student1**.



- 2) Select **Run Now**. Click **OK** twice.



f. Observe grid execution in Platform RTM.

- 1) Restore Platform RTM. If you are prompted, log on as **Student** with a password of **Metadata0**.

- 2) Change the status to **ACTIVE**.
- 3) Click **go** several times to observe the status of the job.

**Batch Job Filters**

Cluster: All User: All UGroup: All Status: **ACTIVE** Effic: (highlighted)

Queue: All Host: All HGroup: All Records: 30 Except:

JobID: Search:  Dynamic **go** clear export

<< Previous Showing Rows 1 to 5 of 5 [1]

Job ID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time
<b>1698</b>	2:\ahmed:C3Flow:c04	RUNNING	2	\ahmed	1s	2.70%	09-07 13:07:23
<b>1699</b>	US_Customers	RUNNING	2	\ahmed	-	-	09-07 13:07:32
<b>1700</b>	US_Customers	RUNNING	2	\ahmed	-	-	09-07 13:07:42
<b>1701</b>	US_Customers	RUNNING	2	\ahmed	-	-	09-07 13:07:52
<b>1702</b>	US_Customers	PEND	1	\ahmed	-	-	-

- g. Click the blue JobID for your job and confirm that the flow used the priority queue.
- h. When the job completes, minimize SAS Management Console and Platform RTM.

**End of Demonstration**



## Exercises

---

### 1. Using SAS Management Console

- a. Log on to SAS Management Console as **Ahmed** with a password of **Student1**.
- b. View the User Manager plug-in, the Data Library Manager plug-in, and the Server Manager plug-in and answer these questions:
  - 1) Click the **Plug-ins** tab. Click **User Manager**. Clear the **Show Groups** and **Show Roles** check boxes at the top of the workspace.

What users are defined in the metadata? \_\_\_\_\_

- 2) Expand the **Data Library Manager** plug-in and select **Libraries**.

What data libraries are defined in the metadata? \_\_\_\_\_

- 3) Expand the **Server Manager** plug-in.

- c. View the folder structure in SAS Management Console.
- d. Click the **Folders** tab. Then click the **Orion Inc.** folder to view its contents.

What types of objects are stored in the Orion Inc. folder? \_\_\_\_\_

### 2. Using the Schedule Manager Plug-in in SAS Management Console

- a. View the properties of the **c3d2** flow.
- b. Schedule the **c3d2** flow for immediate execution.
- c. Observe grid execution in Platform RTM or SAS Management Console.
- d. Answer these questions:
  - 1) How many jobs are included in **c3d2**? \_\_\_\_\_
  - 2) What are their names? \_\_\_\_\_
- e. Minimize Platform RTM and SAS Management Console.

**End of Exercises**

## 3.2 Solutions

---

### Solutions to Exercises

#### 1. Using SAS Management Console

- Log on to SAS Management Console as **Ahmed** with a password of **Student1**.
- View the User Manager plug-in, the Data Library Manager plug-in, and the Server Manager plug-in and answer these questions:
  - Click the **Plug-ins** tab. Click **User Manager**. Clear the **Show Groups** and **Show Roles** check boxes at the top of the workspace.

What users are defined in the metadata? **Aaron, Ahmed, Anita, Barbara, Bruno (or any of the other users listed)**

<input checked="" type="checkbox"/> Show Users	<input type="checkbox"/> Show Groups	<input type="checkbox"/> Show Roles
User, Group, or Role ▲		
<ul style="list-style-type: none"> <li> Aaron</li> <li> Ahmed</li> <li> Anita</li> <li> Barbara</li> <li> Bruno</li> <li> Bryan</li> <li> Chris</li> <li> Corey</li> <li> Ellen</li> <li> Eric</li> <li> George</li> <li> Gloria</li> </ul>		

- Expand the **Data Library Manager** plug-in and select **Libraries**.

What data libraries are defined in the metadata? **Orion, SASApp - SASDATA, SASApp - wrsdist, SASApp - wrstemp, STP Samples**

Name	Engine
Orion	BASE
SASApp - SASDATA	BASE
SASApp - wrsdist	BASE
SASApp - wrstemp	BASE
STP Samples	BASE

3) Expand the **Server Manager** plug-in.

Name
SASApp - Logical SAS DATA Step Batc...
SASApp - Logical Grid Server
SASApp - Logical OLAP Server
SASApp - Logical Pooled Workspace S...
SASApp - Logical Stored Process Server
SASApp - Logical Workspace Server
SASApp - Logical Connect Server
SASApp - Logical SAS Java Batch Server

- c. View the folder structure in SAS Management Console.
- d. Click the **Folders** tab. Then click the **Orion Inc.** folder to view its contents.

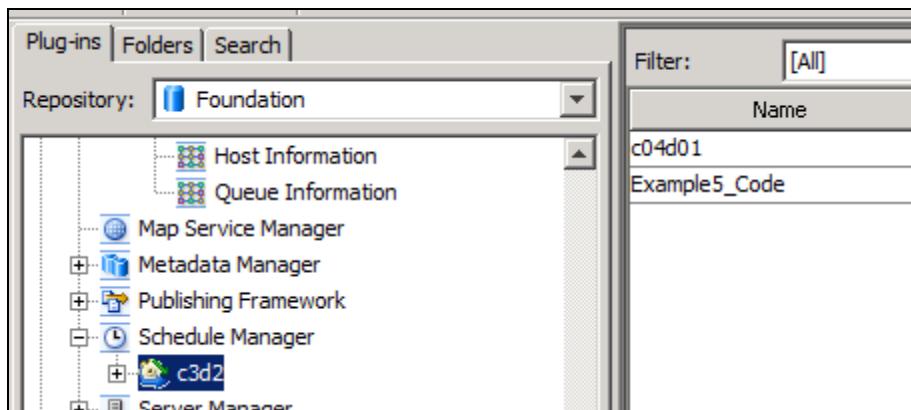
Name	Description	Type
c04d01		Deployed job
c04d01		Job
c04d02		Deployed job
c04d02		Job
CITY	City	Table
CONTINENT	Continent	Table
COUNTRY	Country	Table
COUNTY		Table
CUSTOMER		Table
CUSTOMER_TYPE	Customer Type	Table

What types of objects are stored in the Orion Inc. folder? **folders, deployed jobs, jobs, libraries, and tables**

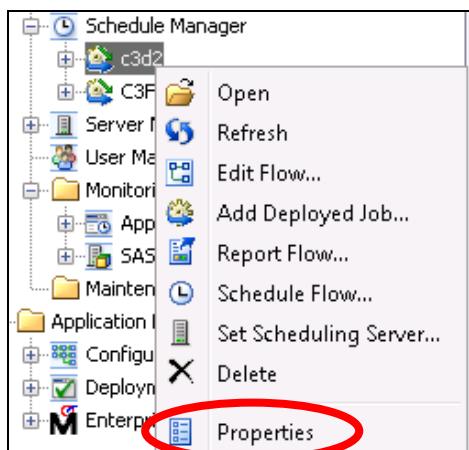
## 2. Using Schedule Manager in SAS Management Console

a. View the properties of the **c3d2** flow.

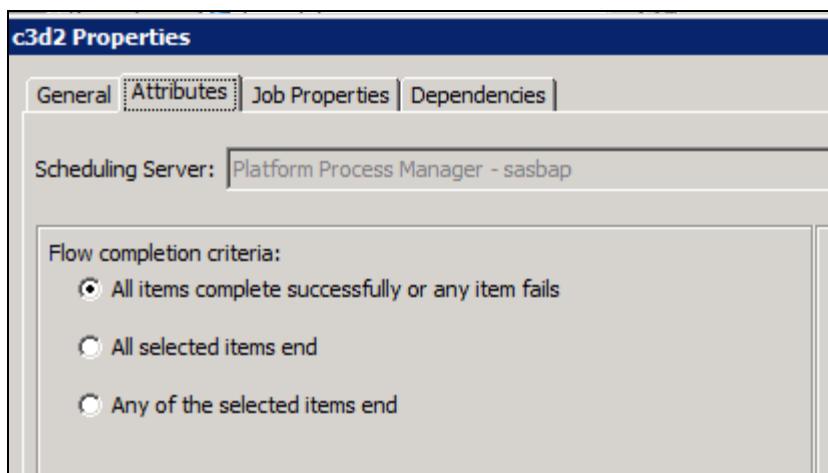
1) Click **c3d2** in Schedule Manager.



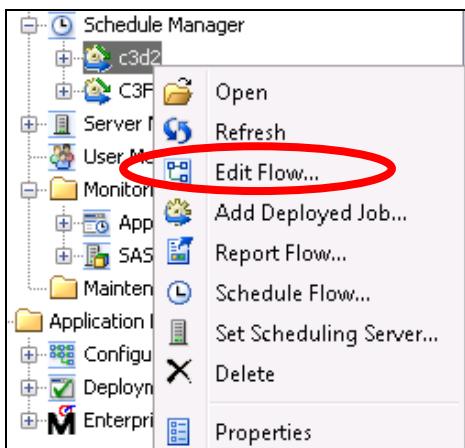
2) Right-click **c3d2** and select **Properties**.



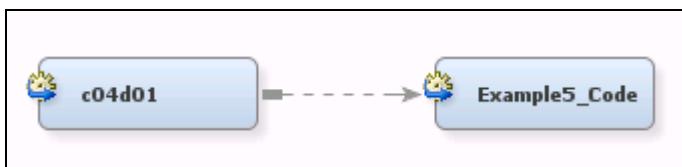
3) Click the **Attributes** tab. Notice the Scheduling Server. Click **Cancel**.



- 4) Right-click **c3d2** and select **Edit Flow**.

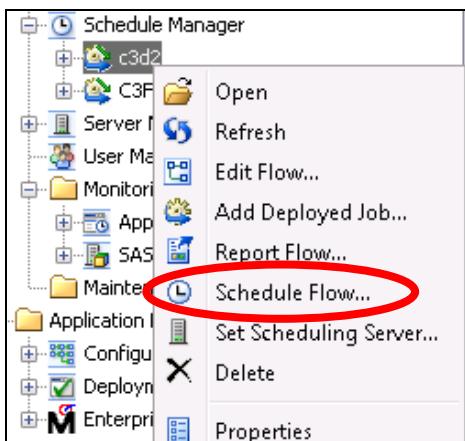


- 5) View the process flow.



- b. Schedule the **c3d2** flow for immediate execution.

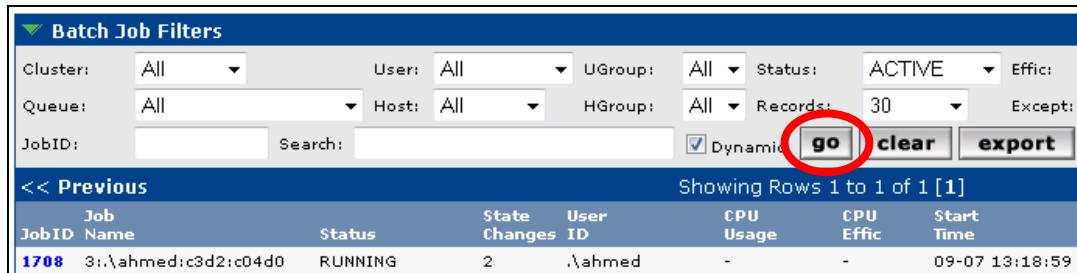
- 1) Right-click **c3d2** and select **Schedule Flow**. If you are prompted, log on as **.\Ahmed** with a password of **Student1**.



- 2) Select **Run Now**  $\Rightarrow$  **OK**.

- c. Observe grid execution in Platform RTM or SAS Management Console.

- 1) In Platform RTM, click the **go** button several times to see the status of the job.



The screenshot shows the 'Batch Job Filters' interface. At the top, there are dropdown menus for Cluster (All), User (All), UGroup (All), Status (ACTIVE), and Effic (All). Below these are Queue (All), Host (All), HGroup (All), Records (30), and Except (None). A checkbox for Dynamic is checked. There are buttons for 'go' (circled in red), 'clear', and 'export'. Below the filters, a message says '<< Previous Showing Rows 1 to 1 of 1 [1]'. A table displays one job entry:

Job ID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time
<b>1708</b>	3:\ahmed:c3d2:c04d0	RUNNING	2	\ahmed	-	-	09-07 13:18:59

- 2) In SAS Management Console, click **Job Information** in Grid Manager to view the status of the job.

d. Answer these questions:

- 1) How many jobs are included in c3d2? **2**
- 2) What are their names? **C04d01 and Example5\_code**

e. Minimize Platform RTM and SAS Management Console.

**End of Solutions**



# Chapter 4    SAS® on the Grid – Working with Code

<b>4.1 SAS Grid Submission in the SAS Windowing Environment .....</b>	<b>4-3</b>
Demonstration: Submitting to the Grid in the SAS Windowing Environment.....	4-24
Exercises .....	4-26
<b>4.2 SAS Grid Enablement with the SCAPROC Procedure .....</b>	<b>4-27</b>
Demonstration: Using the SCAPROC Procedure .....	4-35
Exercises .....	4-38
<b>4.3 SAS Grid Submission Using the SASGSUB Utility.....</b>	<b>4-39</b>
Demonstration: Using SASGSUB .....	4-46
Exercises .....	4-48
<b>4.4 Solutions .....</b>	<b>4-49</b>
Solutions to Exercises .....	4-49
Solutions to Student Activities.....	4-55



## 4.1 SAS Grid Submission in the SAS Windowing Environment

### Objectives

- Describe changes to code that enable you to execute your entire SAS program or portions of your SAS program on the grid from the SAS windowing environment.
- Explain sequential job execution on the grid.
- Describe parallel processing to further exploit the grid.

3

### Executing Programs on a SAS Grid in the SAS Windowing Environment

In its most basic form, using a SAS grid requires adding only five lines of code to an existing program. The code does the following:

1. enables the grid
2. signs on to the grid
3. submits a program to a grid session
4. ends the submission
5. signs off the grid

4

## SAS Code Grid Syntax

```

Example2 Code.sas
/*
/* 1. enable Grid resource */
%let rc=%sysfunc(grdsvc_enable(_all_,resource=SASApp))

/*
/* 2. signon to Grid */
signon session1;

/*
/* 3. remote submit to Grid */
rsubmit session1;

data a;
  input x y z;
  cards;
1 2 3
4 5 6
7 8 9
;
run;
proc summary data=a;
proc summary data=a;
proc summary data=a;

/*
/* 4. end remote submit */
endrsubmit;

/*
/* 5. signoff Grid */
signoff session1;

```

5

It takes a minimum of the following five SAS language statements to make your SAS program run on the grid:

- Enable the grid
- Signon
- Remote Submit
- End Remote Submit
- Signoff

The code above lists the exact syntax. When the GRDSVC\_ENABLE() function (grid enable function) is executed, it connects to the metadata server. You can include the following lines to connect to the metadata server:

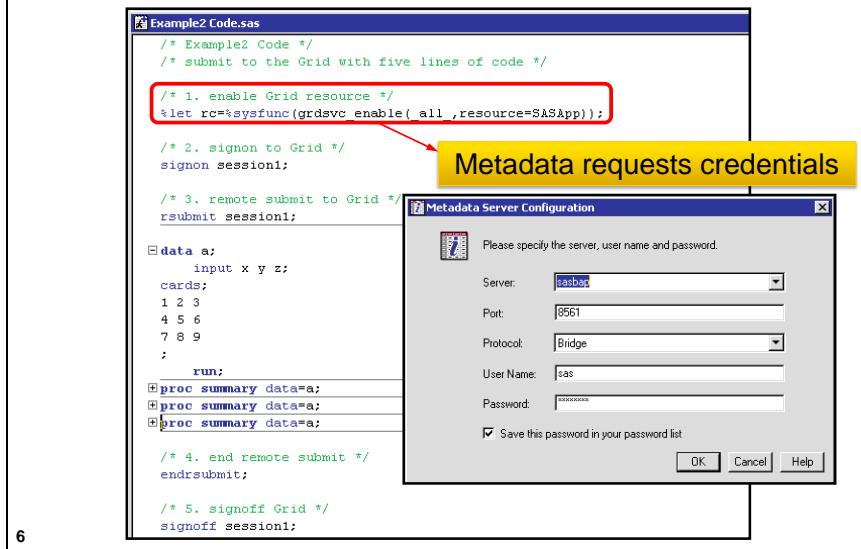
```

options metaserver='server.example.com';
/* where server.example.com is replaced with the server */
/* where your metadata server is running.
options metaport=8561;
options metauser=<user ID>;
options metapass=<encrypted password>;

```

In many situations, your SAS administrator sets up the metadata server connection information for you.

## Sequential SAS Code Grid Syntax



6

If the options for connecting to the SAS Metadata Server are not specified when the `GRDSVC_ENABLE()` function runs, then SAS prompts you for the required information.

## SAS Code Grid Syntax: `grdsvc_enable()`

```
%let rc=%sysfunc(grdsvc_enable
  (_all_,server=SASApp));
```

Element	Definition
<code>%LET rc=</code>	Creates the macro variable <code>rc</code> and assigns it the return code
<code>%SYSFUNC( )</code>	SAS macro used to call SAS functions in open code
<code>GRDSVC_ENABLE( )</code>	SAS grid-enable function
<code>_ALL_ or identifier</code>	<code>_ALL_</code> specifies all signons that use grid sessions. <code>Identifier</code> specifies that signs using it use a grid session
<code>SERVER=SASApp</code>	Specifies which SAS Application Server Context to use for connecting to the grid

7

## SAS Code Grid Syntax: grdsvc\_enable

```
%let rc=%sysfunc(grdsvc_enable(<identifier>,
                                server=SASApp));
```

- This code sends a request to the SAS Metadata Server to enable grid processing for future signons.
- Only sessions for the <identifier> are enabled for grid processing. <identifier> can be a single label or \_ALL\_ to mark all signons as grid enabled.

 The value specified for identifier must be 8 characters or fewer and must start with a character.

8

## SAS Code Grid Syntax: SIGNON Statement

```
signon <identifier>;
```

- This code sends a request to SAS Grid Manager to start a grid session.
- <identifier> is used to refer to the grid session.

9

Successful running of the SIGNON statement results in a SAS grid session running on one of the grid nodes and connected to your SAS session that submitted the SIGNON statement. The grid session runs as the user who submitted the SIGNON statement, which is the user that you are signed in as on your client.

## SAS Code Grid Syntax: Remote Submit

```
rsubmit <identifier>;
  <SAS language statements>
endrsubmit;
```

- This code is referred to as a *remote submission block*.
- It sends the enclosed SAS code to the grid session referred to by *<identifier>*.

10

The code contained in the remote submit block is sent to the grid session where it is run.

## SAS Code Grid Syntax: SIGNOFF Statement

```
signoff <identifier>;
```

- This code sends a request to the grid session associated with the *<identifier>* to end the session.

11

Successful running of the SIGNOFF statement shuts down the grid session on the grid node.

## SAS Code to Grid: Log

The screenshot shows a SAS Log window titled "Log - (Untitled)". The log output is as follows:

```

1  /* Example2 Code */
2  /* submit to the Grid with five lines of code */
3
4  /* 1. enable Grid resource */
5  %let rc=%sysfunc(grdsvc_enable(_all_,resource=SASApp)); ← Red arrow
6
7  /* 2. signon to Grid */
8  signon session1; ← Red arrow
NOTE: Remote session ID SESSION1 will use the grid service ALL .
NOTE: Remote signon to SESSION1 complete.
9
10 /* 3. remote submit to Grid */
11 rsubmit session1;
NOTE: Remote submit to SESSION1 commencing.

1  data a;
2  input x y z;
3  cards;
NOTE: The data set USER.A has been created.
NOTE: DATA statement used:
      real time           0.01 seconds
      cpu time            0.01 seconds

20  proc summary data=a;
21    var z;
22    output out=new3 mean=mz;
23    run;

NOTE: There were 3 observations read from the data set USER.A.
NOTE: The data set USER.NEW3 has 1 observations and 3 variables.
NOTE: PROCEDURE SUMMARY used (Total process time):
      real time           0.01 seconds
      cpu time            0.01 seconds

24
25  /* 4. end remote submit */
NOTE: Remote submit to SESSION1 complete.
12
13  /* 5. signoff Grid */
14  signoff session1;
NOTE: Remote signoff from SESSION1 commencing.
NOTE: Remote signoff from SESSION1 complete.

```

Red arrows highlight specific log entries: one pointing to the "enable Grid resource" step, another to the "signon to Grid" step, and two others pointing to the "remote submit to Grid" and "end remote submit" steps.

12

## 4.01 Poll

The SIGNON statement submits a job to the grid.

- True
- False

13

## SAS Grid Syntax: Specifying Metadata Access

Metadata information can be declared to avoid prompting at execution.

```
options metaserver='sasbap';
options metaport=8561;
options metauser=sasdemo;

%let rc= %sysfunc(grdsvc_enable(_ALL_,
server=SASApp));

signon session1;
```

15

## SAS Grid Syntax: Enable the Grid

The GRDSVC\_ENABLE function enables or disables the grid for one or all SAS sessions.

**GRDSVC\_ENABLE(*identifier* <, *option1*; *option-n*>)**

The value for *identifier* specifies one or more server sessions to be enabled for grid execution.

Optional arguments include the following:

- SERVER= specifies the SAS Application Server that is defined in the metadata.
- WORKLOAD= specifies a value defined in the metadata for the grid middleware to use in selecting grid nodes.
- JOBNAMES= specifies the name of the macro variable that holds the value for the name of the job.
- JOBOPTS= specifies the name of the macro variable that holds the value additional job options.

16

In the GRDSVC\_ENABLE function, notice that there is a comma after the identifier and a semicolon that separates the options.

**GRDSVC\_ENABLE(*identifier*,  
                  *option1*;  
                  *option2*)**

## SAS Grid Syntax: Enable the Grid

```
/* declare metadata information to avoid prompting */
options metaserver='sasbap';
options metaport=8561;
options metauser=sasdemo;
options metapass='XXXXXX';

%let rc = %sysfunc(grdsvc_enable(_all_,
    resource=SASApp;
    jobname=job-name-macro-variable;
    jobopts=job-opts-macro-variable) );
```

17

## SAS Grid Syntax: Specifying Job Names

Job names can be set in SAS code.

```
%let jnvar=FinanceSummary;

%let rc= %sysfunc(grdsvc_enable(_ALL_,
    resource=SASApp;
    jobname=jnvar));

signon session1;
```

18



Options specified in metadata override those specified in a GRDSVC\_ENABLE statement.

## Job Name Displayed in the SAS Grid Manager Plug-in

The screenshot shows the SAS Management Console interface. On the left is a tree view of management services, including Environment Management, Foundation Services Manager, and Grid Manager. Under Grid Manager, there are three sub-options: Grid Monitoring Server - d21552, Job Information, Host Information, and Queue Information. The main pane displays a table titled 'Job Information'. The columns are Job ID /, Job Name, User Name, Status, and Submit Time. There is one row with the values: Job ID / is 1, Job Name is 06FinanceSummary, User Name is CARYNT\dohaig, Status is Running, and Submit Time is Jan 15, 2010 8:17:55 AM.

19

## SAS Grid Syntax: Specifying Job Options

Job options can be set in SAS code.

```
%let jnvar=FinanceSummary;
%let joivar=queue=priority;

%let rc=%sysfunc(grdsvc_enable(_ALL_,
                     server=SASApp,
                     jobname=jnvar,
                     jobopts=joivar));

signon session1;
```

20

## Priority Queue Shown in the SAS Grid Manager Plug-in

The screenshot shows the SAS Management Console interface. On the left, there's a navigation tree with 'Grid Manager' expanded, showing 'Job Information', 'Host Information', and 'Queue Information'. On the right, a table lists various queues:

#	Queue Name	Status	Total Jobs	Pending Jobs	Running Jobs
4	chkpt_rerun_queue	Active	0	0	0
8	die	Active	0	0	0
6	license	Active	0	0	0
3	night	Active	0	0	0
7	normal	Active	0	0	0
1	owners	Active	0	0	0
2	priority	Active	1	1	0
5	short	Active	0	0	0

21

## 4.02 Poll

The JOBOPTS option in the GRDSVC\_ENABLE function enables a user to override permissions set by an administrator.

- True
- False

22

## Moving Data between Sessions

Data might need to be moved between sessions for the program to run correctly. The following three options are available for doing this:

- INHERITLIB option
- UPLOAD and DOWNLOAD procedures

or...

As part of the architecture and configuration of your SAS Grid Manager environment, the administrators can make data and libraries available.

24

## The INHERITLIB Option

The INHERITLIB option enables libraries that are defined in the client session to be inherited by the server session for Read and Write access.

As an option, each client libref can be associated with a libref that is named differently in the server session.

If the server libref is omitted, the client libref name is used in the server session.

25



Many of the statements discussed in this section are documented in the *SAS/CONNECT® 9.4 User's Guide*.

## The INHERITLIB Option

General form of the INHERITLIB option:

```
INHERITLIB=(client-libref1<=server-libref1>
           ...
client-librefn<=server-librefn>)
```

The INHERITLIB option can be placed in the **RSUBMIT** or **SIGNON** statements.

26

## INHERITLIB Option

Example:

```
libname Orion 'c:\mydata\orion\data';
rsubmit p1 WAIT=NO INHERITLIB=(Orion);
  data orion.result;
    /* DATA step statements */
    run;
endrsubmit;
```

This example shows that the libref named **Orion** in the client session is inherited for use in the server session.

27

The remote session uses remote library services to access the data in the inherited library.

- ☞ See the SAS/CONNECT documentation for details about remote library services.

## Transferring the Macro Variable

To transfer the value of the macro variable on the remote session to your local (parent) session, use the %SYSRPUT statement.

General form of the %SYSRPUT statement:

```
%SYSRPUT localmvar = &remotemvar;
```

The %SYSRPUT statement takes the **value** of the macro variable on the **remote** session and places the value into a macro variable in the **local (parent)** session.

28

## Transferring the Macro Variable

To transfer the value of the macro variable on the local session to your remote session, use the %SYSLPUT statement.

General form of the %SYSLPUT statement:

```
%SYSLPUT remotemvar = &localmvar;
```

The %SYSLPUT statement takes the **value** of the macro variable on the **local** session and places the value into a macro variable in the **remote** session.

29

## SYSPUTSYNC Option

By default, macro variables created with the %SYSPUT statement are **not** transferred to the parent session during asynchronous execution until specific execution points.

To transfer the macro variable created with the %SYSPUT statement, you can specify either of the following:

- the SYSPUTSYNC=YES option in the existing RSUBMIT statement
- a new RSUBMIT statement with the WAIT=YES option

30

## The UPLOAD Procedure

The UPLOAD procedure transfers data from the **client (local)** machine to the **server (remote)** machine.

General form of the UPLOAD procedure:

```
PROC UPLOAD <options> <STATUS=NO>;  
RUN;
```

The UPLOAD procedure is always remote submitted.

The STATUS=NO option prevents the status window from appearing during data transfer.

31

# The DOWNLOAD Procedure

The DOWNLOAD procedure transfers data from the ***server (remote)*** machine to the ***client (local)*** machine.

General form of the DOWNLOAD procedure:

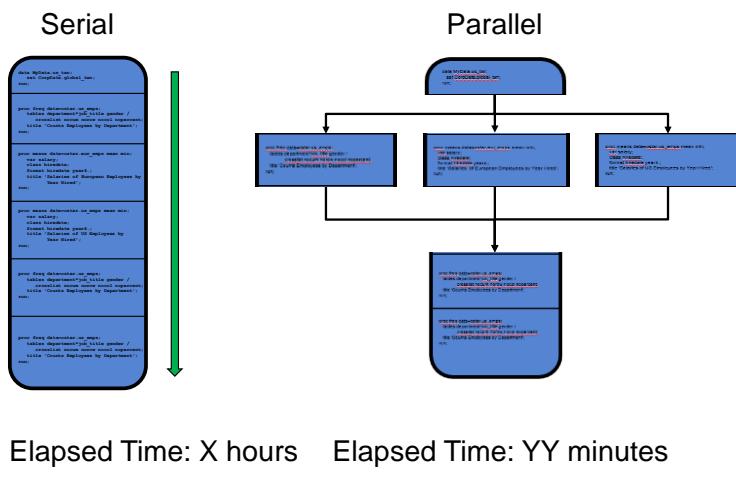
**PROC DOWNLOAD <options> <STATUS=NO>;  
RUN;**

The DOWNLOAD procedure is always ***remote*** submitted.

The options for the UPLOAD and DOWNLOAD procedures are virtually identical.

32

## Parallelized SAS Program



In a SAS environment, SAS programs are run as one serial program. You can think of the SAS program as running the SAS statements one after another from top to bottom. In this case, even if you have segments that are independent, they must wait until all the steps above are completed before they can run.

In a SAS Grid Manager environment, your SAS programs that have independent steps can be run in parallel across the servers in the grid. This greatly reduces the elapsed clock time that it takes your SAS programs to run. Depending on the circumstances, you might be able to take a current SAS program and reduce its processing time from hours to minutes.

The key point is that you can run independent sections of your SAS program in parallel not only in threads on one server, but also in parallel on multiple servers. Many times, this significantly decreases the running time of your SAS program.

## Running Parallel Sessions in the Grid

Using the SAS language statements, you can run sections of your SAS program in multiple grid sessions running at the same time or in parallel.

How do you do it?



34

## Running Parallel Sessions in the Grid

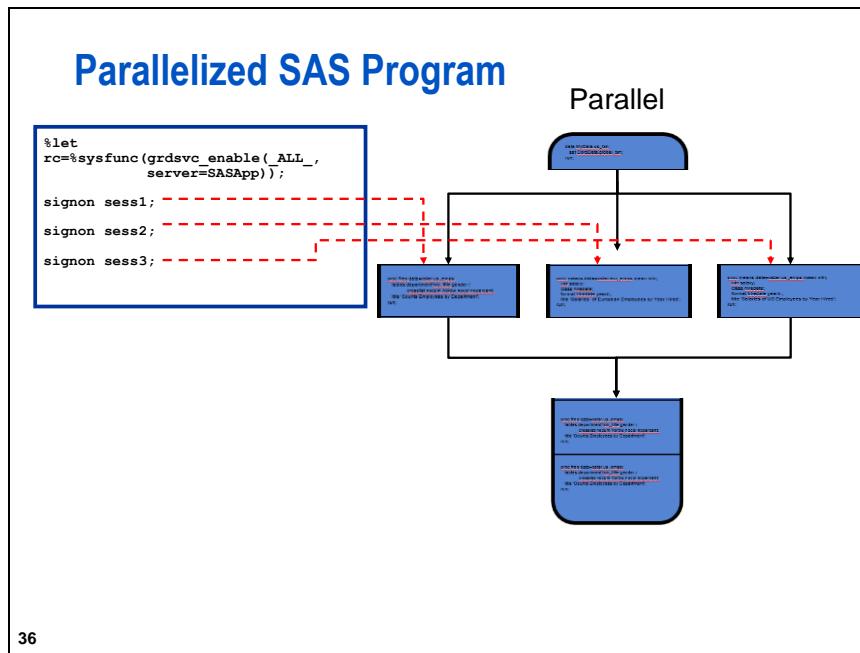
Using the SAS language statements, you can run sections of your SAS program in multiple grid sessions running at the same time or in parallel.

How do you do it?

1. Start multiple SAS Grid Sessions.
  - Each `signon unique_identifier;` line in your code will start a new SAS Grid Session

35

Assuming your code has the grid enable function (`grdsvc_enable()`) and specified that all sessions will be grid sessions then each signon statement run results in a grid session. When starting multiple grid sessions using signon statements you must specify a unique identifier as a parameter to the signon statement. You use the unique identifier to refer to the grid session started by the signon statement.



36

For example, let's say you want to start three grid sessions so you can run blocks of code in parallel as shown in the above diagram. To do this you would do the following:

- Add the statements to call the grid enable function using the `_ALL_` keyword as the first parameter.
- The `_ALL_` keyword flags your session so that each signon will request a grid session
- Add three signon statements each with a unique identifier. In the diagram above you can see that each signon statement starts a grid session.

## Running Parallel Sessions in the Grid

Using the SAS language statements, you can run sections of your SAS program in multiple grid sessions running at the same time or in parallel.

How do you do it?

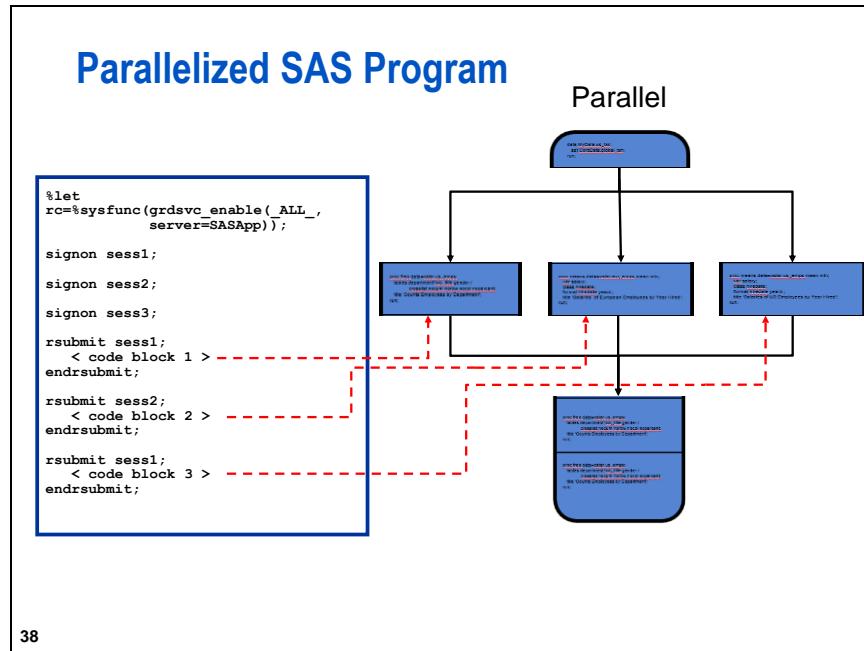
1. Start multiple SAS Grid Sessions.
  - Each `signon unique_identifier;` line in your code will start a new SAS Grid Session.
2. Submit sections of your code to each grid session.
  - Surround the code to run in the grid session with a remote submit block:

```

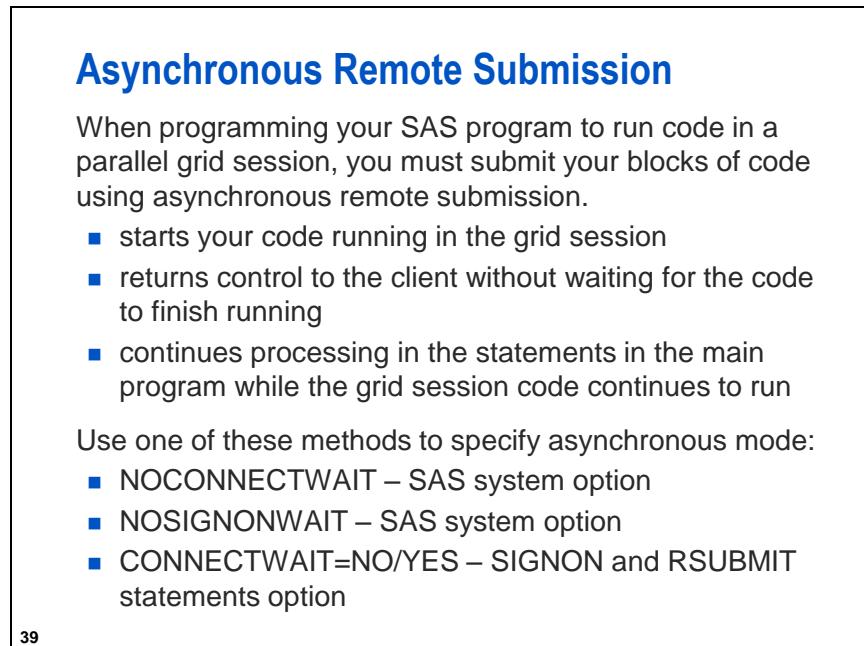
rsubmit unique_identifier;
  <your SAS code>
Endrsubmit;

```

37



To continue our example, you now want to send the three blocks of code that will run in parallel to the three grid sessions. You do this by adding the rsubmit statement before the block of code and add the endrsubmit statement at the end of the block of code. You must add the unique identifier to inform the rsubmit command which grid session to use.



By default, the rsubmit statements submit code to run in the remote session using synchronous mode. This means that your program will submit the remote submit block of code to run in the remote session and wait until the remote code is finished running before returning to your program to run the next statement. When you are running blocks of code in parallel you want to start the remote block of code running and return to your program to start the next block of code. To do this you specify one of the Asynchronous options.

## Controlling Processing: WAITFOR Statement

The WAITFOR statement causes the client session to wait for the completion of one or more tasks that are in progress. It is used with asynchronous submissions.

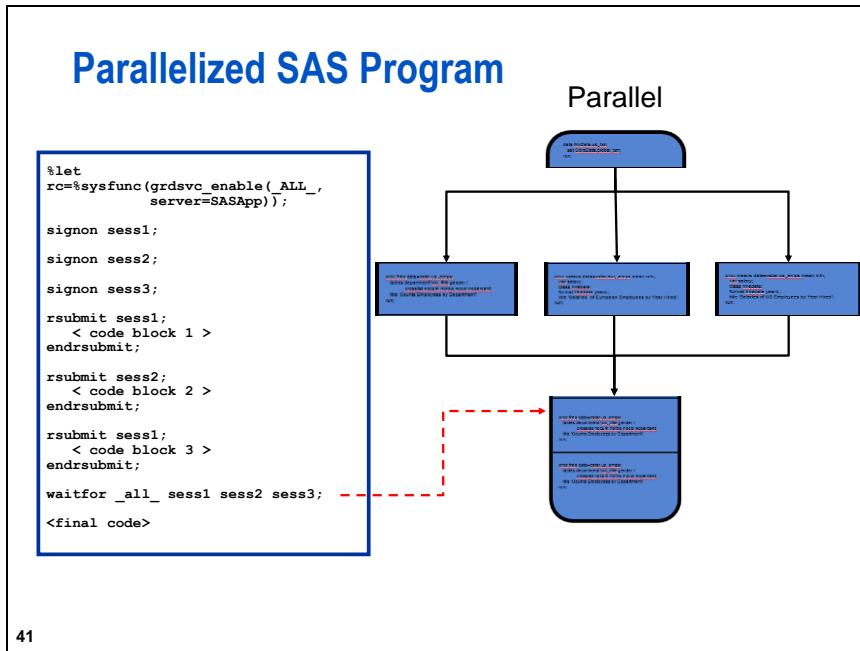
```
WAITFOR <_ANY_|_ALL_> task-1 task-2... <TIMEOUT=x>;
```

- `_ANY_` – waits for any of the tasks to complete.
- `_ALL_` – waits for all of the tasks to complete.
- `TIMEOUT` – After *x* seconds if the tasks are not completed, the WAITFOR statement terminates.

40

For the times where you have blocks of code running in parallel you will have code after these blocks of code that needs to wait to run after the blocks of parallel code is finished running. In other words, you need your program to wait until the previous blocks of code are finished running before continuing with the rest of the statements in your program.

You use the waitfor statement to control further processing in your program.



41

Back to our example. Before you run the last sections of your program you want to wait until the parallel blocks of code have finished running. To do this you add the `waitfor` statement:

`waitfor _all_ sess1 sess2 sess3;`

This instructs your program to wait at this line until all of the sessions listed after the `_all_` keyword have finished before continuing to run the statements in your program after this statement.

You could use:

`waitfor _any_ sess1 sess2 sess3;`

If you only want to wait until one of the listed sessions is finished before continuing.

## SAS Grid Syntax: Parallel Processing

```
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp));  
signon session1; ← Session 1  
rsubmit session1 wait=yes;  
/* session1 code to be remote submitted */  
endrsubmit;  
signon session2; ← Session 2  
rsubmit session2 wait=no;  
/* session2 code to be remote submitted */  
endrsubmit;  
signon session3; ← Session 3  
rsubmit session3 wait=no;  
/* session3 code to be remote submitted */  
endrsubmit;  
signon session4; ← Session 4  
rsubmit session4 wait=no;  
/* session4 code to be remote submitted */  
endrsubmit;  
waitfor _all_ session1 session2 session3 session4;  
/* more submit blocks or SAS code */  
signoff _all_;
```

42



## Submitting to the Grid in the SAS Windowing Environment

This demonstration illustrates how to submit jobs to the grid in the SAS windowing environment.

1. Start SAS by selecting **Start**  $\Rightarrow$  **All Programs**  $\Rightarrow$  **SAS**  $\Rightarrow$  **SAS 9.4 (English)**, or use the shortcut on the desktop.
2. Open **c04d01.sas** by navigating to **C:\Grid\Grid Demo Docs\SAS Code on Grid**.
  - a. Click **File**  $\Rightarrow$  **Open Program**.
  - b. Navigate to **C:\grid\Grid Demo Docs\SAS Code on Grid**.
  - c. Select **c04d01.sas**  $\Rightarrow$  **Open**.
  - d. Submit the program. Click the **Submit** icon or select **Run**  $\Rightarrow$  **Submit**.
3. When you are prompted, enter **sasbap** as the server, **Nancy** as the user name, and **Student1** as the password. Click **OK** to submit credentials.

The screenshot shows the SAS Log window titled "Log - (Untitled)" running on an "X64\_S08R2 WIN 6.1.7601 Service Pack 1 Server". The log output includes:

```

Log - (Untitled)
X64_S08R2 WIN 6.1.7601 Service Pack 1 Server

NOTE: SAS initialization used:
      real time      5.41 seconds
      cpu time       0.71 seconds

NOTE: AUTOEXEC processing beginning; file is D:\Program
      Files\SASHome\SASFoundation\9.4\autoexec.sas.

NOTE: Libref ORFMTS was successfully assigned as follows:
      Engine:      V9
      Physical Name: D:\Workshop\OrionStar\orfmt

NOTE: AUTOEXEC processing completed.
1  /* c04d01.sas */
2  /* submit to the Grid with five lines of code */
3
4  /* 1. enable Grid resource */
5  %let rc=%sysfunc(grdsvc_

```

A "Metadata Server Configuration" dialog box is overlaid on the log window. It contains fields for Server (set to "sasbap"), Port (set to "8561"), Protocol (set to "Bridge"), User Name (set to "Nancy"), and Password (set to "xxxxxx"). A checked checkbox says "Save this password in your password list". At the bottom are "OK", "Cancel", and "Help" buttons.

The bottom of the log window shows the command being submitted:

```

c04d01.sas  Processing submitted status
/* c04d01.sas */
/* submit to the Grid with five lines of code */

/* 1. enable Grid resource */

```

4. Observe the job that executes in either Platform RTM or SAS Management Console. Use the command line tools at a command prompt.

View the job in Platform RTM for SAS. Due to the short run time of this program and how often Platform RTM for SAS polls for data, you might need to change the status in the Batch Job Filters window at the top of the web page from ACTIVE to **DONE**.

The screenshot shows the 'Batch Job Filters' interface. The search criteria are set to 'All' for Cluster, User, UGroup, Queue, Host, HGroup, and Status (ACTIVE). The 'Records' dropdown is set to 30. The 'JobID' search field contains '1818'. The results table shows one row:

Job ID /	Job Name	Status	User Changes	User ID	CPU Usage	CPU Effic	Start Time	Pend
1818	SASGrid:6912	PEND	1	\Student	-	-	-	1m

View the job in SAS Management Console.

#	Job ID /	Job Name	User Name	Status	Submit Time	Start Time	End Time	Execute Ho...
1	1818	SASGrid:6912	\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap

The status varies depending on when the views are refreshed.

Open a command window and view the jobs using the **bjobs -u all** command.

5. In SAS, open **c04d02.sas** from the same directory.
- Submit the program. Click the **Submit** icon or select **Run**  $\Rightarrow$  **Submit**.
6. Observe the job executing in either Platform RTM or SAS Management Console.

View in Platform RTM

The screenshot shows the 'Batch Job Filters' interface. The search criteria are set to 'All' for Cluster, User, UGroup, Queue, Host, HGroup, and Status (ACTIVE). The 'Records' dropdown is set to 30. The 'JobID' search field contains '1819'. The results table shows two rows:

Job ID /	Job Name	Status	User Changes	User ID	CPU Usage	CPU Effic	Start Time
1819	US_Customers	RUNNING	2	\Student	-	-	09-13 09:56:16
1820	US_Customers	RUNNING	2	\Student	-	-	09-13 09:56:31

View in SAS Management Console

#	Job ID /	Job Name	User Name	Status	Submit Time	Start Time	End Time	Execute Ho...
2	1818	SASGrid:6912	\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap
1	1819	US_Customers	\Student	Running	Sep 13, 201...	Sep 13, 201...		sasbap

The status varies depending on when the views are refreshed.

Open a command window and view the jobs using the **bjobs -u all** command.

**End of Demonstration**



## Exercises

---

### 1. Submitting to the Grid from the SAS Windowing Environment

- a. If you did not follow along with the demonstration, start SAS by selecting **Start** ⇒ **All Programs** ⇒ **SAS** ⇒ **SAS 9.4 (English)**.
- b. Run **c04e01.sas**, which can be found in **C:\Grid\Grid Demo Docs\SAS Code on Grid**. View results in Platform RTM and SAS Management Console. Use the command line tools at a command prompt.
- c. Modify **c04e01.sas** to automatically sign on to the metadata server as **Student**.
- d. Modify the program to include a job name, **Curr\_emps**.
- e. Modify the program to use the priority queue.
- f. Observe the job executing in either Platform RTM or SAS Management Console. Use the command line tools at a command prompt.

**End of Exercises**

## 4.2 SAS Grid Enablement with the SCAPROC Procedure

### Objectives

- Describe the SCAPROC procedure.
- Explain the syntax of the SCAPROC procedure.
- List special requirements for SAS libraries.

46

### SCAPROC Procedure

- The SCAPROC procedure initiates the use of the SAS Code Analyzer.
- PROC SCAPROC is a Base SAS procedure.
- PROC SCAPROC executes an existing SAS program, and records, analyzes, and generates commented SAS code.
- PROC SCAPROC helps convert existing SAS applications to run in parallel steps on the grid.
- Resulting code can be
  - executed on the SAS grid
  - imported into SAS Data Integration Studio
  - used in your programs.

47



Additional information is available at <http://www2.sas.com/proceedings/forum2008/006-2008.pdf>. (*Introducing the SAS® Code Analyzer* is a relevant paper by Eric Thies and Rick Langston.)

## Using PROC SCAPROC

1. Activate PROC SCAPROC.
  - Designate the file locations to which results should be written.
2. Submit your existing SAS program.
  - Notice that the program ***fully executes***.
3. Close the SCAPROC process.
  - Use the SCAPROC WRITE statement.

The results are two files that contain special comments.

- Code Analyzer output with comments
- grid-exploiting code with remote submit blocks

48

The existing program runs completely within the PROC SCAPROC wrapper. Both results are fully executable SAS programs.

## The SCAPROC Procedure

General form of the SCAPROC procedure:

```

PROC SCAPROC;
  RECORD 'RECORD.SAS'
  GRID 'GRECORD.TXT';
RUN;

<SAS program to be analyzed>

PROC SCAPROC;
  WRITE;
RUN;

```

49

## Using PROC SCAPROC: Syntax

```

proc scaproc;
  record '\\SASBAP\.....\Comments_out.txt'
    /* Code Analyzer Comments */
    grid '\\SASBAP\...\Grid_out.txt'
    /* Grid Exploiting Code */
  resource "SASApp"; /* SAS Server */
run;

proc summary data=wk_data.a;
  var x; output out=new1 mean=mx; run;
proc summary data=wk_data.a;
  var y; output out=new2 mean=my; run;
proc summary data=wk_data.a;
  var z; output out=new3 mean=mz; run;

proc scaproc;
  write; /* Closes the results files */
run;

```

50

In activation syntax, designate two locations for output and the name of your SAS server.

### 4.03 Poll

Using PROC SCAPROC with the RECORD statement submits a job for execution.

- Yes
- No

51

## Results: Code Analyzer Comments

Partial PROC SCAPROC Results

**Job information**

```
/* JOBSPLIT: DATASET OUTPUT SEQ WK_DATA.A.DATA */
/* JOBSPLIT: LIBNAME WK_DATA V9
   '\\SASBAP\SAS_Data\Wk_data' */
/* JOBSPLIT: PROCNAME DATASETP */
/* JOBSPLIT: DATASET OUTPUT SEQ WK_DATA.NEW1.DATA */

/* JOBSPLIT: ELAPSED 267 */
/* JOBSPLIT: PROCNAME SUMMARY */
/* JOBSPLIT: STEP SOURCE FOLLOWS */
proc summary data=wk_data.a;
  var x;
  output out=wk_data.new1 mean=mx;
run;
...
```

**End of job**

53

## Results: Grid-Exploiting Code

Partial Results for Grid-Exploiting Code

**Job information**

```
-----*/
/* There are 4 tasks in this job. */
/* 4 of these tasks can be RSUBMITTED. */
/* These 4 tasks used 581 units of time. */
/* The longest task took 267 units of time,
   46% of total time. */

-----*/
```

**Select grid**

```
/* Enable grid service */
%let rc=%sysfunc(grdsvc_enable(_all_,
  resource=SASApp));
-----*/

/* This function call initializes data
   structures for our SCAGRID functions. */
proc scaproc; startup 4 &SCAGRID_SESSIONS;
run;
```

54

*continued ...*

## Results: Grid-Exploiting Code

Establish grid sessions.

**Loop to signon to grid sessions**

```
%macro scagrid_sessions(count);
  %do i = 1 %to &count;
    signon sess&i connectwait=no inheritlib=(user);
    %put Session started on grid node
    %sysfunc(grdsvc_getname(sess&i));
  %end;
%mend scagrid_sessions;
```

55

## Results: Grid-Exploiting Code

Submit blocks of code to grid sessions.

**Loop through remote submit tasks**

```
proc scaproc; getsession 1 "sess"; run;
%put sess=&sess;
/*-----*/
/* rsubmit for task 1 */
/*-----*/
rsubmit &sess sysrputsync=yes
cmacvar=scagrid_task_2;
proc summary data=wk_data.a; var x;
  output out=wk_data.new1 mean=mx; run;
endrsubmit;
```

The code repeats a remote submit block for all job tasks.

- ✍ Verify that all required libraries are set up in each remote submission block. You might need to add libraries to the remote submission blocks.

56

In production grid environments, the SAS administrator might set up most libraries for you either as metadata libraries, in an autoexec.sas file, or as a predefined library.

## Using PROC SCAPROC

Any data used by the grid must be ***permanently available to all nodes***. Recommendations for best practice include the following:

1. Administrators can declare all librefs in advance in the user-modifiable autoexec\_usermods.sas program or use predefined libraries.
2. Use a staging library to share data between sessions.
  - User level
  - Group level

57

continued...

## Using PROC SCAPROC

Any data used by more than one grid must be ***permanently available to all nodes***. Recommendations for best practice include the following:

3. Change the ***implied*** Work library reference to permanent librefs. This converts **dsname** to **User.dsname**.

```
option User='\\MachineName\SharedFolder\...';
```

58

## Using PROC SCAPROC: Permanent Data

59

```

Example1 SCAPROC.sas
/* Example1 SCAPROC */
/* contains explicit permanent file locations */
libname wk_data '\\SASBAP\SAS_Data\Wk_data';

proc scaproc;
  record '\\SASBAP\SCAPROC Output\Ex1_Comments_out.txt' /*Code Analyzer Comments*/
  grid '\\SASBAP\SCAPROC Output\Ex1_Grid_out.txt' /* Grid Exploiting Code */
  resource "SASApp"; /* SAS Server */
run;

data wk_data.a;
  input x y z;
cards;
1 2 3
4 5 6
7 8 9
;
run;

proc summary data=wk_data.a;
  var x;
  output out=wk_data.new1 mean=mx;
run;

proc summary data=wk_data.a;
  var y;
  output out=wk_data.new2 mean=my;
run;

proc summary data=wk_data.a;
  var z;
  output out=wk_data.new3 mean=mz;
run;

proc scaproc;
  write;
run;

```

**Data is stored permanently via a libref other than Work. It is also identified in appserver\_autoexec\_usermods.sas.**

## Using PROC SCAPROC: USER= Option

60

```

Log - (Untitled)
77 /* Example 2 SCAPROC */
78 /* Implied WORK, two level dataset names */
79 /* converted to USER.dsnames that are permanent */
80 libname Wk_Data '\\SASBAP\SAS_Data\Wk_data';
NOTE: Libname WK_DATA refers to the same physical library as USER.
NOTE: Libref WK_DATA was successfully assigned as follows:
  Engine: V9
  Physical Name: \\SASBAP\SAS_Data\Wk_data
81 Options user='\\SASBAP\SAS_Data\Wk_data';
82
83 data a;
84   input x y z;
85 cards;

NOTE: The data set USER.A has 3 observations and 3 variables.
NOTE: DATA statement used (total process time):
  real time      0.03 seconds
  cpu time       0.01 seconds

```

This is the log that results when you execute PROC SCAPROC grid output. It shows the following:

- the use of autoexec USER= *code*
- grid start-up
- code submissions

## Using PROC SCAPROC: Resulting Output

```

/*
 * There are 4 tasks in this job
 * 4 of these tasks can be RSUBMITTED
 * These 4 tasks used 201 units of time
 * The longest task took 78 units of time, 38.8% of total time.
 */
/*
 * This is the user-modifiable number of connect sessions
 * Numbers of sessions should be between 1 and 20 inclusive.
 */
/*
 * Enable grid service
 %let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp));

```

61

After you submit the program for execution, the resulting files are created.

The named output location can use a .txt suffix, but using the .sas suffix enables the code to be executed without further modification.

The grid-exploiting output contains this line:

```
%let scagrid_sessions=4;
```

This is a default setting by the PROC SCAPROC developers. You can change it manually in the resulting program.

## Observing Grid Execution in SAS Management Console

#	Job ID	Job Name	User Name	Status	Submit T...	Start Time	End Time	Execut...	Queue
1	1760	SASGrid:3748.sas	sas	Pending	Nov 18, 200...				normal
6	1759	SASGrid:3748.sas	sas	Done	Nov 18, 200...	Nov 18, 200...	Nov 18, 200...	sasbi	normal
5	1758	SASGrid:3748.sas	sas	Done	Nov 18, 200...	Nov 18, 200...	Nov 18, 200...	sasbi	normal
4	1757	SASGrid:4624.sas	sas	Done	Nov 18, 200...	Nov 18, 200...	Nov 18, 200...	sasbi	normal
3	1756	SASGrid:4624.sas	sas	Done	Nov 18, 200...	Nov 18, 200...	Nov 18, 200...	sasbi	normal
2	1755	SASGrid:4624.sas	sas	Done	Nov 18, 200...	Nov 18, 200...	Nov 18, 200...	sasbi	normal

62



## Using the SCAPROC Procedure

1. If necessary, start SAS by selecting **Start**  $\Rightarrow$  **All Programs**  $\Rightarrow$  **SAS**  $\Rightarrow$  **SAS 9.4 (English)**.
  - a. Click **File**  $\Rightarrow$  **Open Program**. Navigate to **C:\Grid\Grid Demo Docs\SCAPROC Demo** and open **c04d03.sas**.
  - b. Submit **c04d03.sas**.

Partial Program

```
/* c04d03.sas */
/* contains explicit permanent file locations */

options metaserver='sasbap';
options metaport=8561;
options metauser='Student'
      metapass='{SAS002}BA7B9D061CB4066E47F2455F373B030E';

proc scaproc;
  record 'C:\Grid\Grid Demo Docs\SCAPROC Demo\SCAPROC Output
          \c04d03_Comments_out.sas' /* Code Analyzer Comments*/
  grid 'C:\Grid\Grid Demo Docs\SCAPROC Demo\SCAPROC Output
        \c04d03_Grid_out.sas'      /* Grid Exploiting Code */
  resource "SASApp";           /* SAS Server */
run;

libname orion meta library="Orion";
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
data ostar.us_cust;
  set orion.customer;
  where country='US';
run;

...
proc freq data=ostar.us_cust;
  tables age*gender gender;
run;

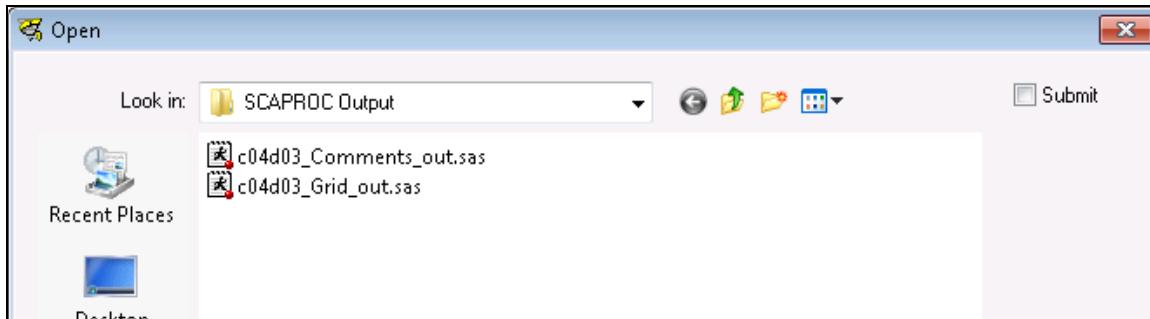
proc means data=ostar.us_cust;
  var age;
  class customer_type_ID;
run;

...
proc scaproc;
  write;
run;
```



No jobs were submitted to the grid.

- c. Open the results folder item **c04d03\_Comments\_out.sas** by navigating to **C:\Grid\Grid Demo Docs\SCAPROC Demo \SCAPROC Output\**.



- d. Review the results for **c04d03\_Comments\_out.sas**.

#### Partial Results

```
/* JOBSPLIT: LIBNAME ORION meta library="Orion" */
/* JOBSPLIT: DATASET OUTPUT SEQ OSTAR.US_CUST.DATA */
/* JOBSPLIT: LIBNAME OSTAR 'C:\Grid\Grid Demo Docs\DI on Grid' */
/* JOBSPLIT: SYMBOL GET SYS IOUSEEE */
/* JOBSPLIT: ELAPSED 121 */
/* JOBSPLIT: PROCNAME DATASTEP */
/* JOBSPLIT: STEP SOURCE FOLLOWS */

libname orion meta library="Orion";
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
data ostar.us_cust;
  set orion.customer;
  where country='US';
run;

/* JOBSPLIT: DATASET INPUT SEQ ORION.CUSTOMER.DATA */
/* JOBSPLIT: LIBNAME ORION meta library="Orion" */
```

- e. Close the Editor window.
2. Open the results folder item **c04d03\_Grid\_out.sas** by navigating to **C:\Grid\Grid Demo Docs\SCAPROC Demo \SCAPROC Output\**.
- a. Review the results and notice the grid-exploiting syntax (**c04d03\_Grid\_out.sas**).

#### Partial Results

```
/*-----*/
/* There are 6 tasks in this job. */
/* 6 of these tasks can be RSUBMITed. */
/* These 6 tasks used 2372 units of time. */
/* The longest task took 1373 units of time, 57.9% of total time. */
/*-----*/
/* This is the user-modifiable number of connect sessions */
/* Numbers of sessions should be between 1 and 20 inclusive. */
/*-----*/
%let SCAGRID_SESSIONS=3;
```

```
/*
/* *** Please don't edit anything below this line. */
/* *** Regenerate the file if you need to make changes. */

/*
/* Enable grid service
*/
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp));
```

- b. Modify **c04d03.sas** by adding this statement to the second remote session that creates the **non\_us** data set:

```
libname orion meta library="Orion";
```

- c. Add this statement to all remote sessions after the first:

```
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
```

 The final version of the program is saved as **C:\Grid\Grid Demo Docs\SCAPROC Demo\SCAPROC Output\c04d03\_Grid\_Mod.sas**.

- d. Submit **c04d03\_Grid\_out.sas** and observe the grid jobs in Platform RTM or SAS Management Console.

#### Platform RTM

Batch Job Filters							
Cluster:	All	User:	All	UGroup:	All	Status:	ACTIVE
Queue:	All	Host:	All	HGroup:	All	Records:	30
JobID:		Search:		<input checked="" type="checkbox"/> Dynamic	<b>go</b>	<b>clear</b>	<b>export</b>
<< Previous				Showing Rows 1 to 1 of 1 [1]			
Job	JobID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic
	1937	SASGrid:7508	RUNNING	2	.\student	-	-
						09-19 09:29:02	

#### SAS Management Console

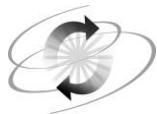
455	SASGrid:1076668	.\student	Done	May 23, 2014 1:28:34 PM	May 23, 2014 1:28:36 PM	May 23, 2014 1:28:44 PM
456	SASGrid:1076668	.\student	Done	May 23, 2014 1:28:34 PM	May 23, 2014 1:28:41 PM	May 23, 2014 1:28:45 PM
457	SASGrid:1076668	.\student	Done	May 23, 2014 1:28:35 PM	May 23, 2014 1:28:46 PM	May 23, 2014 1:28:48 PM
458	SASGrid:1076668	.\student	Done	May 23, 2014 1:28:35 PM	May 23, 2014 1:28:51 PM	May 23, 2014 1:28:52 PM

#### Command prompt

C:\Users\student>bjobs -u all							
JOBID	USER	STAT	QUEUE	FROM_HOST	EXEC_HOST	JOB_NAME	SUBMIT_TIME
455	student	RUN	normal	sasbap	sasbap	*d:1076668	May 23 13:28
456	student	RUN	normal	sasbap	sasbap	*d:1076668	May 23 13:28
457	student	PEND	normal	sasbap	sasbap	*d:1076668	May 23 13:28
458	student	PEND	normal	sasbap	sasbap	*d:1076668	May 23 13:28

3. Close the Editor window.

**End of Demonstration**



## Exercises

---

### 2. Exploiting the Grid with the SCAPROC Procedure

- a. If necessary, start SAS by selecting Start  $\Rightarrow$  All Programs  $\Rightarrow$  SAS  $\Rightarrow$  SAS 9.4 (English).
- b. Run **c04e02.sas**, which is found in C:\Grid\Grid Demo Docs\SCAPROC Demo\.
- c. Open the resulting program, **c04e02\_Grid\_out.sas**, in C:\Grid\Grid Demo Docs\SCAPROC Demo\SCAPROC Output\.
- d. Notice the grid-exploiting syntax.
- e. Modify **c04e02\_Grid\_out.sas** to include the appropriate LIBNAME statements and submit the program.

Hint: The second RSUBMIT block should include these two statements:

```
libname orion meta library="Orion";
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
```

and the following RSUBMIT blocks should include this statement:

```
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
```

Observe the grid jobs in Platform RTM, SAS Management Console, or use the command line tools.

- f. Further modify **c04e02\_Grid\_out.sas**.
  - 1) Change the number of sessions in the program to five. Submit and observe the job in Platform RTM.
  - 2) Modify the GRDSVC\_ENABLE call to include a job name. Submit and observe the job in Platform RTM.

**End of Exercises**

## 4.3 SAS Grid Submission Using the SASGSUB Utility

### Objectives

- Describe the SAS Grid Manager Client Utility, also known as SASGSUB.
- Explain the use of SASGSUB.

66

### What Is SASGSUB?

SASGSUB is a stand-alone utility that enables users to do the following:

- submit batch SAS jobs to the grid in synchronous or asynchronous modes
- display the status of jobs on the grid
- stream back the log
- retrieve output from jobs to a local directory
- kill submitted jobs
- on a UNIX or Linux grid, start SAS in interactive modes on a grid node
  - Line mode
  - Display Manager mode – requires an X server on the client

67

## Benefits of SASGSUB

- A user can submit a job and forget it when he uses asynchronous mode.
  - There is no need to remain connected to process a job.
- A user can view the job output while the job is running.
- SASGSUB enables the SAS checkpoint/restart capability.
- SASGSUB uses SAS Grid Manager metadata for centralized control.
- A user can script the conditional submission of SAS batch jobs using synchronous mode.

68

## SASGSUB: Submit Program Syntax

- From the command line, use this command:

```
sasgsub -gridsubmitpgm <sas_pgm>
```
- Other arguments to SASGSUB can be stored in the configuration file or added to the command. They include the following:
  - -GRIDWORK <shared\_file\_dir>
  - <metadata\_connection\_parameters>
  - -GRIDAPPSERVER <app\_server\_name>
  - -GRIDSASOPTS *grid-sas-options*
- The SAS program is submitted to the grid and control is immediately returned to the client. This is referred to as *asynchronous submission*.

69

## SASGSUB: Submit Program Syntax

- From the command line, use this command:

```
sasgsub -gridsubmitpgm <sas_pgm>
         -gridwaitresults
```

- The SAS program is submitted to the grid and control is not returned to the client until the SAS program finishes running. This is referred to as *synchronous submission*.
- The return code from the SAS program is returned from the SASGSUB.
- Synchronous grid batch submission is commonly used when using third-party schedulers.

70

## SASGSUB: Getting Job Status

To obtain the job status, use this command:

```
sasgsub -gridgetstatus <job_id | _ALL_>
```

### Sample Output

```
Current Job Information
Job 1917 (testPgm) is Finished: Submitted: 08Dec2008:10:28:57,
                           Started: 08Dec2008:10:28:57 on Host d15003,
                           Ended: 08Dec2008:10:28:57
Job 1918 (testPgm) is Finished: Submitted: 08Dec2008:10:28:57,
                           Started: 08Dec2008:10:28:57 on Host d15003,
                           Ended: 08Dec2008:10:28:57
Job 1925 (testPgm) is Submitted: Submitted: 08Dec2008:10:28:57
```

71

## SASGSUB: Retrieving Results

To retrieve results, use this command:

```
sasgsub -gridgetresults <job_id | _ALL_>
```

### Sample Output

```
Current Job Information
Job 1917 (testPgm) is Finished: Submitted:
 08Dec2008:10:53:33, Started: 08Dec2008:10:53:33
  on Host d15003, Ended: 08Dec2008:10:53:33
  Moved job information to .\SASGSUB-2008-11-
  21_21.52.57.130_testPgm

Job 1918 (testPgm) is Finished: Submitted:
 08Dec2008:10:53:33, Started: 08Dec2008:10:53:33
  on Host d15003, Ended: 08Dec2008:10:53:33
  Moved job information to .\SASGSUB-2008-11-
  24_13.13.39.167_testPgm

Job 1925 (testPgm) is Submitted: Submitted:
 08Dec2008:10:53:34
```

72

## SASGSUB: Killing a Job

Other parameters are stored in a configuration file.

```
sasgsub -gridkilljob <job_id | _ALL_>
```

### Sample Output

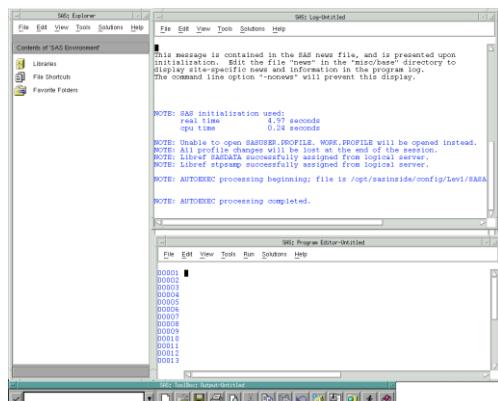
```
Termination requested for job ID 6773.word
```

73

## SASGSUB: Start Interactive Display Manager on the Grid

At a command prompt, run the following:

```
sasgsub -gridrunsasdms client:0
```



74

## 4.04 Multiple Choice Poll

Arguments to the SASGSUB command include which of the following?

- a. GRIDSUBMITPGM
- b. GRIDGETRESULTS
- c. GRIDKILLJOB
- d. GRIDRUNSASDMS
- e. all of the above

75

## SASGSUB: Results Directory Setup

SASGSUB creates a folder for each received submission.

```
<Directory user invoked sasgsub with the GETJOBRESULTS option>
|
+--SASGSUB-YYYY-MM-DD_HH:MM_SS_mmm<job_name>
|
+--<SAS_program>.sas
+--<SAS_program>.lst
+--<SAS_program>.log
+--job.info
+--<input_files>
+--<output_files>
+--WORK
```

Job.info contains the following:

Job ID:	<job_id>
Job Name:	<job_name>
Time job submitted:	<submit_time>
Time job started:	<start_time>
Time job completed:	<end_time>
Name of execution host:	<host_name>

77

<b>&lt;SAS_program&gt;.sas</b>	Created by the client program; SAS program specified by user.
<b>&lt;SAS_program&gt;.lst</b>	Created by SAS; SAS list file.
<b>&lt;SAS_program&gt;.log</b>	Created by SAS; SAS log file.
<b>job.info</b>	Created by the client program. This file contains the job submit, start, and end time along with the execution host and job ID.
<b>&lt;input_files&gt;</b>	Created by the client program. Files are staged using the GRIDFILESEN option.
<b>&lt;output_files&gt;</b>	Created by SAS. Other files are created by a program and placed in the GRIDWORK library.
<b>Work</b>	Created if GRIDRESTARTOK is specified; contains SAS restart information.

## SASGSUB Get Results

sasgsub –gridgetresults <Job ID>

SAS Grid Submit Utility Version 9.40 (build date: Dec 4 2013)  
 Copyright (C) 2009-2013, SAS Institute Inc., Cary, NC, USA. All Rights Reserved

432 (classlist) is Finished:  
 Submitted: 23May2014:09:02:54,  
 Started: 23May2014:09:03:33 on Host SASBAP,  
 Ended: 23May2014:09:04:21, RC:0  
 Moved job information to ".\SASGSUB-2014-05-23\_09.02.16.576\_classlist"

```
Directory of C:\Users\student\SASGSUB-2014-05-23_09.02.16.576_classlist
05/23/2014  09:13 AM    <DIR>      .
05/23/2014  09:13 AM    <DIR>      ..
05/23/2014  09:03 AM            2,192 classlist.log
05/23/2014  09:03 AM            1,978 classlist.lst
05/23/2014  09:00 AM            36 classlist.sas
05/23/2014  09:13 AM            220 job.info
                           4 File(s)       4,426 bytes
```

78

## SASGSUB: Observed Execution in RTM

The screenshot shows the SAS Grid Management interface. The top navigation bar includes 'config', 'graphs', 'alerts', 'grid' (which is highlighted in orange), and 'HA'. A sidebar on the left has sections for 'Dashboards', 'Management' (which is also highlighted in orange), and 'Job Info'. Under 'Job Info', there are links for 'By Host', 'By Host Group', 'By Project', 'By Queue', 'By Array', and 'Details'. The main area displays a table of batch jobs with columns: JobID\*, Name, Status, State Changes, User ID, and CPU Usage. Job ID 4703, named 'correx1', is circled in red.

JobID*	Name	Status	State Changes	User ID	CPU Usage
4703	correx1	DONE	1	\sas	0.05s
4702	correx1	DONE	0	\sas	0.02s
4701	SASGrid:6468	DONE	2	\sas	0.05s
4700	SASGrid:6468	DONE	2	\sas	0.03s
4699	SASGrid:6468	DONE	2	\sas	0.03s

79



## Using SASGSUB

---

1. View the program **c04d04.sas**.
  - a. If necessary, start SAS.
  - b. Click **File** ⇒ **Open Program**.
  - c. Navigate to **C:\grid\Grid Demo Docs\SASGSUB demo**.
2. Open **c04d04.sas**.
  - a. Use SASGSUB to run the program on the grid as a batch program.
    - a) Open a command prompt using the shortcut on the desktop.
    - b) Change directories to **cd C:\grid\Grid Demo Docs\SASGSUB demo**.
    - c) Submit the program **c04d04.sas** using SASGSUB. Run this command:

```
sasgsub -gridsubmitpgm c04d04.sas
```

```
C:\grid\Grid Demo Docs\SASGSUB demo>sasgsub -gridsubmitpgm c04d04.sas
SAS Grid Submit Utility Version 9.40 (build date: Dec 4 2013)
Copyright (C) 2009-2013, SAS Institute Inc., Cary, NC, USA. All Rights Reserved
Job <459> is submitted to default queue <normal>.
Job ID: 459
Job directory: "\sasbap\gridwork\student\SASGSUB-2014-05-23_14.34.22.009_c04d04"
Job log file: "\sasbap\gridwork\student\SASGSUB-2014-05-23_14.34.22.009_c04d04\c04d04.log"
```

- b. Observe execution on the grid from the command prompt by entering the **bjobs -u all** command.
- c. Retrieve the results using this command: **sasgsub -gridgetresults 459**.



Use your job ID as shown when you used SASGSUB to submit your program.

```
C:\grid\Grid Demo Docs\SASGSUB demo>sasgsub -gridgetresults 459
SAS Grid Submit Utility Version 9.40 (build date: Dec 4 2013)
Copyright (C) 2009-2013, SAS Institute Inc., Cary, NC, USA. All Rights Reserved
 459 (c04d04) is Finished: Submitted: 23May2014:14:34:22, Started: 23May2014:14:34:30 on Host SASB
AP, Ended: 23May2014:14:34:31, RC:0
  Moved job information to ".\SASGSUB-2014-05-23_14.34.22.009_c04d04"
```

3. View the results.
  - a. In the command prompt window, verify that the directory specified in the output message from SASGSUB exists by entering the **dir** command. Press Enter.
  - b. In the command prompt window, change to the following directory that holds the output:

```
cd SASGSUB-2014-05-23_14.34.22.009_c04d04
```

- c. Open the output file in Notepad by entering this command at the command prompt:

**notepad c04d04.lst**

Press Enter.

- d. View the output. Close Notepad.

**End of Demonstration**



## Exercises

---

### 3. Using the SASGSUB Utility to Submit SAS Programs to the Grid

- a. Review the program **c04e03.sas**.
- b. Submit the program to the grid using SASGSUB.
- c. Retrieve the results.
- d. View the output and log files.

**End of Exercises**

# 4.4 Solutions

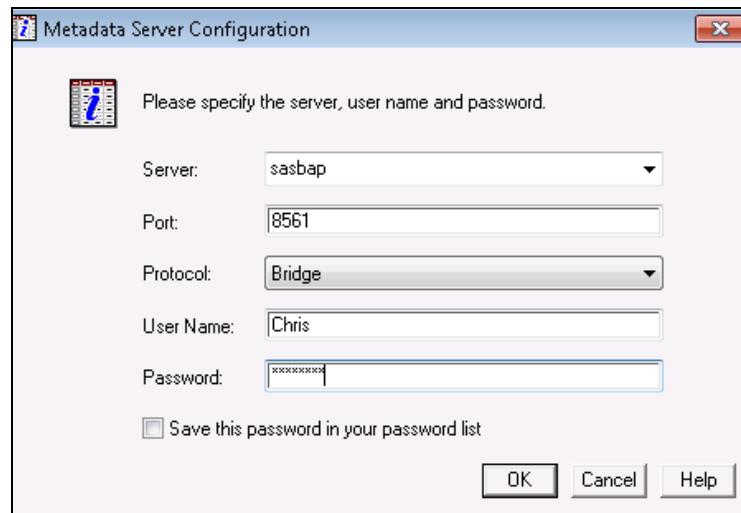
---

## Solutions to Exercises

### 1. Submitting to the Grid from the SAS Windowing Environment

- a. If you did not follow along with the demonstration, start SAS by selecting **Start** ⇒ **All Programs** ⇒ **SAS** ⇒ **SAS 9.4 (English)**.
- b. Run **c04e01.sas**, which can be found in **C:\Grid\Grid Demo Docs\SAS Code on Grid**. Use the command line tools at a command prompt.
  - 1) Click **File** ⇒ **Open Program**.
  - 2) Navigate to **C:\Grid\Grid Demo Docs\SAS Code on Grid**.
  - 3) Select **c04e01.sas** and click **Open**.
  - 4) Submit the program. Click the **Submit** icon or select **Run** ⇒ **Submit**.

 If you did not follow along with the demonstration, you are prompted to sign on.



- 5) View the results in Platform RTM and SAS Management Console. Use the command line tools at a command prompt.

#### Platform RTM Results

<< Previous		Showing Rows 1 to 9 of 9 [1]					
Job ID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time
1826	SASGrid:6912	DONE	0	\student	0.2s	2.54%	09-13 10:10:33
1825	SASGrid:6912	DONE	0	\student	0.23s	1.30%	09-13 10:10:22
1824	SASGrid:6912	DONE	0	\student	0.19s	0.69%	09-13 10:10:12

## SAS Management Console Results

#	Job ID	Job Name	User Name	Status	Submit Time	Start Time	End Time	Execute Ho...
9	1826	SASGrid:6912	.\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap
8	1825	SASGrid:6912	.\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap
7	1824	SASGrid:6912	.\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap
6	1823	SASGrid:6912	.\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap

- c. Modify **c04e01.sas** to automatically sign on to the metadata server as **Student**.

- 1) Add the last line of the following code to the beginning of the program:

```
/* predeclare metadata information to avoid prompting */
options metaserver='sasbap';
options metaport=8561;
options metauser='Student' metapass='Metadata0';
```

- 2) Submit the program and observe it in either monitor.

- d. Modify the program to include a job name, **Curr\_emps**.

- 1) Add the following code to the beginning of the program:

```
/* set Job Name job options */
%let jnvar=Curr_emps;

/* enable Grid resources */
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp;
                                jobname=jnvar));
```

 Your job name might be different.

- 2) Submit the program and observe it in either monitor.

- e. Modify the program to use the priority queue.

- 1) Modify the code as follows:

```
/* set Job Name and Priority Queue job options */
%let jnvar=Curr_emps;
%let jovar=queue=priority;

/* enable Grid resources */
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp;
                                jobname=jnvar; jobopts=jovar));
```

The complete final program is in **c04s01.sas**.

- f. Submit the final program and view it in Platform RTM, SAS Management Console, and using the command line tools at a command prompt.

The views should resemble the following:

#### Platform RTM

Batch Job Filters								
Cluster:	All	User:	All	UGroup:	All	Status:	ACTIVE	Effic:
Queue:	All	Host:	All	HGroup:	All	Records:	30	Except:
JobID:		Search:			<input checked="" type="checkbox"/> Dynamic	go	clear	export
<< Previous		Showing Rows 1 to 3 of 3 [1]						
Job ID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time	
1827	Curr_emps	RUNNING	2	\student	-	-	09-13 10:20:43	
1828	Curr_emps	RUNNING	2	\student	-	-	09-13 10:20:53	
1829	Curr_emps	PEND	1	\student	-	-	-	

#### SAS Management Console

#	Job ID	Job Name	User Name	Status	Submit Time	Start Time	End Time	Execute Ho...
3	1829	Curr_emps	\Student	Pending	Sep 13, 201...			
2	1828	Curr_emps	\Student	Running	Sep 13, 201...	Sep 13, 201...		sasbap
1	1827	Curr_emps	\Student	Running	Sep 13, 201...	Sep 13, 201...		sasbap
12	1826	SASGrid:6912	\Student	Done	Sep 13, 201...	Sep 13, 201...	Sep 13, 201...	sasbap

#### Command Line Tools

C:\Users\student>b_jobs -u all							
JOBID	USER	STAT	QUEUE	FROM_HOST	EXEC_HOST	JOB_NAME	SUBMIT_TIME
560	student	RUN	priority	sasbap	sasbap	Curr_emps	May 27 10:46
561	student	RUN	priority	sasbap	sasbap	Curr_emps	May 27 10:46
562	student	RUN	priority	sasbap	sasbap	Curr_emps	May 27 10:46

## 2. Exploiting the Grid with the SCAPROC Procedure

- a. If necessary, start SAS by selecting **Start**  $\Rightarrow$  **All Programs**  $\Rightarrow$  **SAS**  $\Rightarrow$  **SAS 9.4 (English)**.
- b. Run **c04e02.sas**, which is found in **C:\Grid\Grid Demo Docs\SCAPROC Demo**.
  - 1) Click **File**  $\Rightarrow$  **Open Program**.
  - 2) Navigate to **C:\Grid Demo Docs\SCAPROC Demo**.
  - 3) Select **c04e02.sas** and click **Open**.
  - 4) Submit the program. Click the **Submit** icon or select **Run**  $\Rightarrow$  **Submit**.
- c. Open the resulting program, **c04e02\_Grid\_out.sas** in **C:\Grid\Grid Demo Docs\SCAPROC Demo\SCAPROC Output**.
  - 1) Click **File**  $\Rightarrow$  **Open Program**.
  - 2) Navigate to **C:\Grid Demo Docs\SCAPROC Demo\SCAPROC Output**.
  - 3) Select **c04e02\_Grid\_out.sas** and click **Open**.

- d. Notice the grid-exploiting syntax **c04e02\_Grid\_out.sas**. Before you run this grid-enabled version of your program, you must make a few changes to make sure each block of code that is remote submitted has the proper libraries set up.
- e. Modify **c04e02\_Grid\_out.sas** to include the appropriate LIBNAME statements and submit it.

The second RSUBMIT block should include these two statements:

```
libname orion meta library="Orion";
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
```

It requires both of these libraries because the code in that block uses both the **orion** and **ostar** libraries.

The remaining RSUBMIT blocks should include this statement:

```
libname ostar 'C:\Grid\Grid Demo Docs\DI on Grid';
```

Observe the grid jobs in Platform RTM, SAS Management Console, or use the command line tools. You should see four grid sessions running for your program.

- f. Further modify **c04e02\_Grid\_out.sas**.

- 1) Change the number of sessions in the program to five. Submit and observe the job in Platform RTM.

- a) Change the value of the **SCAGRID\_SESSIONS** macro variable to **5**.

```
/*-----*/
/* This is the user-modifiable number of connect sessions */
/* Numbers of sessions should be between 1 and 20 inclusive.*/
/*-----*/
%let SCAGRID_SESSIONS=5;
```

- b) Submit the program.
- c) Observe the job in Platform RTM.

Batch Job Filters							
Cluster:	All	User:	All	UGroup:	All	Status	
Queue:	All	Host:	All	HGroup:	All	Records	
Presets:	Custom	From:	2011-05-01 09:52	To:	2011-05-02 09:52	2	
JobID:		Search:		<input checked="" type="checkbox"/> Dynamic	go	clear	export
<< Previous							
Showing Rows 1 to 21 of 21 [1]							
Job		Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time
JobID**	Name						
5168	SASGrid:18944	PEND	1	\sas	-	-	-
5167	SASGrid:18944	RUNNING	2	\sas	-	-	05-02 11:14:12
5166	SACGW:110044	DONE	3	\sas	0.05	0.04%	05-02 10:54:27

- 2) Modify the GRDSVC\_ENABLE call to include a job name. Submit and observe the job in Platform RTM.

- a) Add a %LET statement and change the GRDSVS\_ENABLE call. (The complete program is in **c04s02\_Grid\_out.sas** in **C:\grid\Grid Demo Docs\SCAPROC Demo\**.)

```
%let jnvar=SCAPROC_job;
```

```
/*
 * Enable grid service
 */
%let jnvar=SCAPROC_Job;
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp,
                                jobname=jnvar));
```

- b) Submit the program.
- c) Observe the job in Platform RTM, the Grid Manager plug-in, or using the command line tools.

Job ID	Name	Status	State Changes	User ID	CPU Usage	CPU Effic	Start Time	Pend
1946	SCAPROC_job	DONE	0	\student	0.2s	2.26%	09-19 10:05:15	0.1m
1945	SCAPROC_job	DONE	0	\student	0.17s	1.01%	09-19 10:05:06	0.1m
1944	SCAPROC_job	DONE	0	\student	0.22s	0.81%	09-19 10:04:55	0.1m

Please refer to the file **c:\grid\Grid Demo Docs\SCAPROC Demo\c04s02\_Grid\_out.sas** for the finished program solution.

### 3. Using the SASGSUB Utility to Submit SAS Programs to the Grid

- a. Review the program **c04e03.sas**
- b. Submit the program to the grid using SASGSUB.
  - 1) In a command window, change to the directory containing the program.
  - 2) Use SASGSUB to submit the program using the following command at the command prompt:

```
sasgsub -gridsubmitpgm c04e03.sas
```

```
SAS Grid Submit Utility Version 9.40 (build date: Dec 4 2013)
Copyright (C) 2009-2013, SAS Institute Inc., Cary, NC, USA. All Rights Reserved
Job <564> is submitted to default queue <normal>
Job ID: 564
Job directory: "\sasbap\gridwork\student\SASGSUB-2014-05-27_11.28.31.632_c04e03"
Job log file: "\sasbap\gridwork\student\SASGSUB-2014-05-27_11.28.31.632_c04e03\c04e03.log"
```

- c. Retrieve the results
    - 1) In the command window use the following command:
 

```
sasgsub -gridgetresults 564
```

 where 564 is your job ID.
- ```
c:\grid\Grid Demo Docs\SASGSU demo>sasgsub -gridgetresults 564
SAS Grid Submit Utility Version 9.40 (build date: Dec 4 2013)
Copyright (C) 2009-2013, SAS Institute Inc., Cary, NC, USA. All Rights Reserved
564 (c04e03) has Failed: Submitted: 27May2014:11:28:32, Started: 27May2014:11:28:39 on Host
AP, Ended: 27May2014:11:28:40, RC:2
Moved job information to ".\SASGSUB-2014-05-27_11.28.31.632_c04e03"
```

d. View the output and log files.

Using Notepad, open the files **c04e03.lst** and **c04e03.log** from the directory that was created when you retrieved the results. It should be a directory whose name is similar to the following: SASGSUB-2014-05-23\_14.34.22.009\_c04e03

```
Directory of c:\grid\Grid Demo Docs\SASGSUB demo
05/27/2014  11:34 AM    <DIR>      .
05/27/2014  11:34 AM    <DIR>      ..
08/31/2011  02:14 PM          2,666 c04e04.sas
05/23/2014  03:05 PM          396 c04e03.sas
05/27/2014  11:30 AM    <DIR>      SASGSUB-2014-05-27_11.28.31.632_c04e03
                           2 File(s)   3,062 bytes
                           3 Dir(s)  13,748,248,576 bytes free
```

```
c:\grid\Grid Demo Docs\SASGSUB demo\SASGSUB-2014-05-27_11.28.31.632_c04e03>dir
Volume in drive C has no label.
Volume Serial Number is 4019-A5F6

Directory of c:\grid\Grid Demo Docs\SASGSUB demo\SASGSUB-2014-05-27_11.28.31.632_c04e03
05/27/2014  11:30 AM    <DIR>      .
05/27/2014  11:30 AM    <DIR>      ..
05/27/2014  11:28 AM          4,220 c04e03.log
05/23/2014  03:05 PM          396 c04e03.sas
05/27/2014  11:30 AM          220 job.info
                           3 File(s)   4,836 bytes
                           2 Dir(s)  13,747,724,288 bytes free
```

**End of Solutions**

## Solutions to Student Activities

### 4.01 Poll – Correct Answer

The SIGNON statement submits a job to the grid.

- True
- False

14

False. The SIGNON statement sends the request for a grid session to SAS Grid Manager, and a grid session is started on one of the grid nodes.

### 4.02 Poll – Correct Answer

The JOBOPTS option in the GRDSVC\_ENABLE function enables a user to override permissions set by an administrator.

- True
- False

23

### 4.03 Poll – Correct Answer

Using PROC SCAPROC with the RECORD statement submits a job for execution.

- Yes
- No

52

### 4.04 Multiple Choice Poll – Correct Answer

Arguments to the SASGSSUB command include which of the following?

- a. GRIDSUBMITPGM
- b. GRIDGETRESULTS
- c. GRIDKILLJOB
- d. GRIDRUNSASDMS
- e. all of the above

76

# Chapter 5     SAS® Client Applications on the Grid

|                                                                                              |             |
|----------------------------------------------------------------------------------------------|-------------|
| <b>5.1 SAS Clients and SAS Workspace Servers .....</b>                                       | <b>5-3</b>  |
| <b>5.2 SAS Data Integration Studio with the SAS Grid.....</b>                                | <b>5-8</b>  |
| Exercises .....                                                                              | 5-14        |
| <b>5.3 SAS Enterprise Guide on the SAS Grid.....</b>                                         | <b>5-17</b> |
| Demonstration: Using SAS Enterprise Guide on the Grid .....                                  | 5-24        |
| Exercises .....                                                                              | 5-29        |
| <b>5.4 SAS Add-In for Microsoft Office Running a SAS Stored Process on the SAS Grid.....</b> | <b>5-30</b> |
| Exercises .....                                                                              | 5-32        |
| <b>5.5 SAS Enterprise Miner on the SAS Grid .....</b>                                        | <b>5-34</b> |
| Exercises .....                                                                              | 5-36        |
| <b>5.6 Solutions .....</b>                                                                   | <b>5-37</b> |
| Solutions to Exercises .....                                                                 | 5-37        |
| Solutions to Student Activities.....                                                         | 5-53        |



# 5.1 SAS Clients and SAS Workspace Servers

## Objectives

- Describe how SAS client applications start SAS Workspace Servers in a SAS Platform environment **without** a grid.
- Describe how SAS client applications start SAS Workspace Servers in a SAS Platform environment **with** a grid.
  - when workspace servers **are not** load balanced
  - when workspace servers **are** load balanced
  - when workspace servers **are** load balanced and using grid-launched workspace servers
- Use SAS Work libraries in a SAS grid.

3

## SAS Clients and SAS Workspace Servers

Each of the following SAS client applications use a SAS Workspace Server for accessing files and data, and for running SAS programs on a server.

- SAS Add-In for Microsoft Office
- SAS Data Integration Studio
- SAS Enterprise Guide
- SAS Enterprise Miner
- SAS Studio

4

## SAS Clients and SAS Workspace Servers

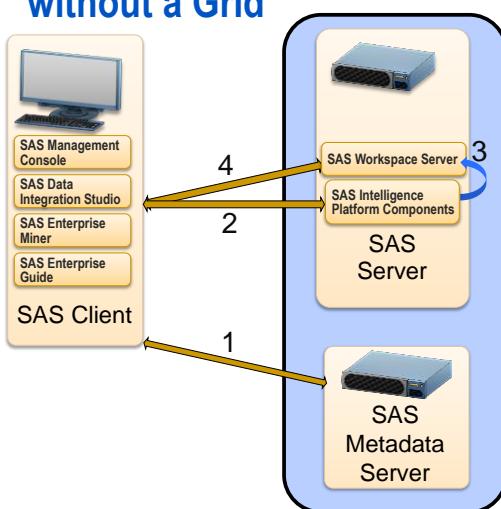
### What is a SAS Workspace Server?

A SAS Workspace Server is an instance of SAS running on a server to support client applications. Features of a SAS Workspace Server are these:

- Runs on a server without a display
- Controlled by a client application
- Used to:
  - run SAS programs
  - access files on the server
  - access any data can access via SAS Libraries
- Runs using the credentials of the client

5

## SAS Clients Requesting a Workspace Server without a Grid



1. Get the location of the object spawner.
2. Request a workspace server.
3. Start the workspace server.
4. Connect to the client.

10

To understand how SAS Workspace Servers and SAS grid sessions work in a SAS grid environment, you must first understand how the client applications start and use a SAS Workspace Server in a SAS Platform environment without a SAS grid.

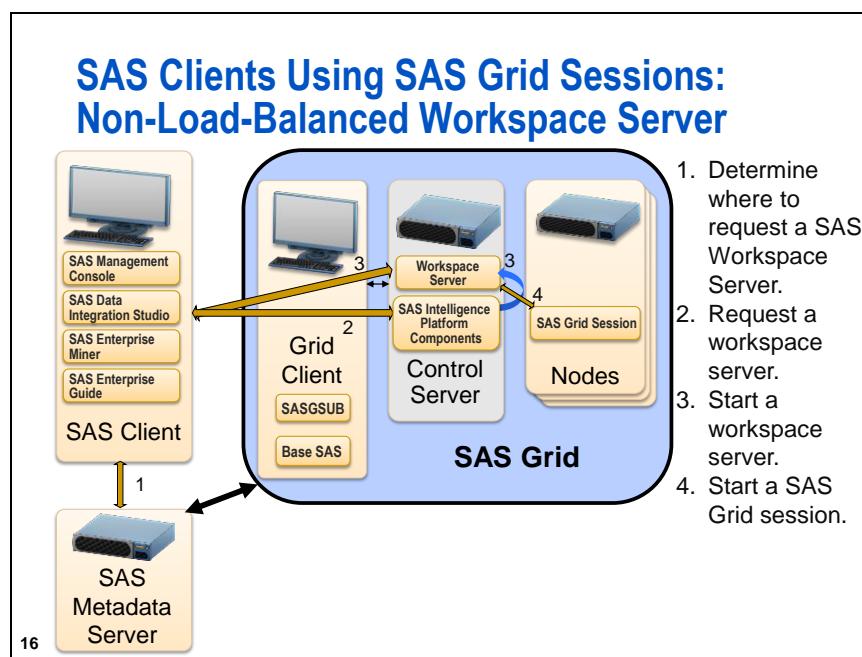
When one of the listed SAS client applications requires SAS to access data or files, or to perform SAS processing (such as running SAS code), the client goes through a series of steps to start a SAS Workspace Server on a server.

In a SAS Platform environment, SAS Intelligence Platform components start all SAS Workspace Servers when it receives a request from a client application.

Steps to start a SAS Workspace Server:

1. The client application requests the location where SAS Intelligence Platform components are running from the SAS Metadata Server. The SAS Metadata Server returns the server name and connection information to the client.
2. The client sends a request to the SAS Intelligence Platform components running on the server returned from the request to the metadata server for a workspace server.
3. The SAS Intelligence Platform components start the workspace server for the client.
4. The client is given the connection to the workspace server.

Now, the client application can use the workspace server for accessing resources and running SAS code on the server. Each client application instance has its own SAS Workspace Server.



SAS Workspace Servers are not set up as load balanced by default. When a workspace server is load balanced, it is configured so that when requests are received for a new workspace server, SAS starts the workspace server on a given set of servers.

The following steps are followed when the client applications request SAS processing on the grid when the workspace servers are **not** load balanced.

1. The client application requests the location where SAS Intelligence Platform components are running from the SAS Metadata Server. The SAS Metadata Server returns the server name and connection information to the client.
2. The client sends a request to the SAS Intelligence Platform components running on the server returned from the request to the metadata server for a workspace server.
3. The SAS Intelligence Platform components start the workspace server for the client, and connects the client to the workspace server.

4. The workspace server then requests a SAS grid session. SAS Grid Manager starts a new grid session on the best available server that the user of the application is allowed to use.

## SAS Clients Using Grid Sessions with a Load-Balanced Workspace Server

- A workspace server is load balanced when it is configured to run on a set of servers.
- The SAS Intelligence Platform components starts the workspace server for the user on one of the servers that the workspace server is load balanced across.
- Grid sessions are still started by the workspace server, but the workspace server runs on one of the set of servers.
- In this configuration, each user will get a workspace server and a grid session. The majority of the work is done by the grid session.

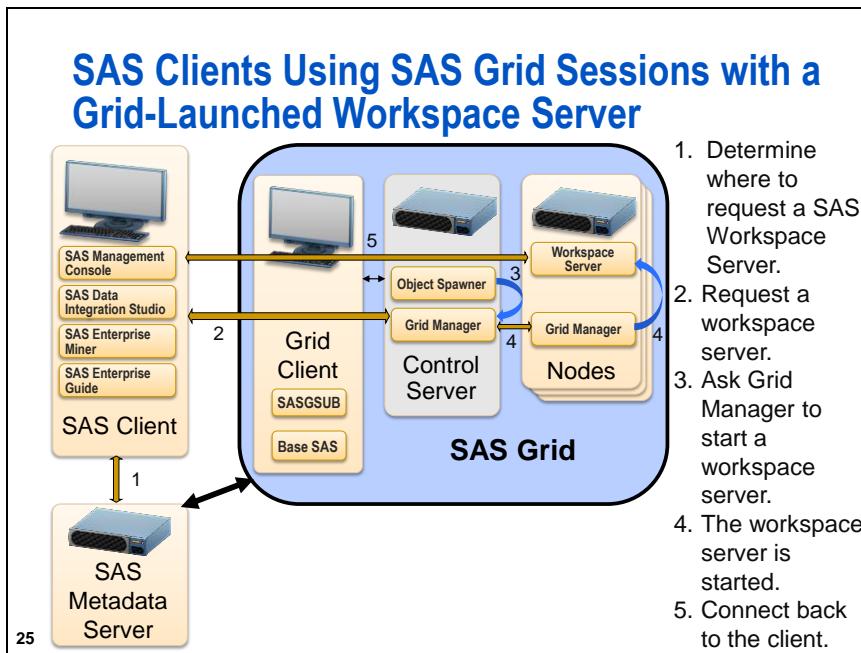
17

## SAS Clients Using Grid Sessions with a Grid-Launched Workspace Server

SAS 9.4 enables SAS Workspace Servers to be started by SAS Grid Manager on one of the servers in the grid.

- SAS Grid Manager starts the workspace server on the server that is least busy.
- It enables client applications to use only a workspace server.
- There is no need for a separate grid session.
- You no longer need to set any of the client application options that specify to “use the grid”.
- Your Administrator can configure SAS to ignore the “use the grid” options.

18



As mentioned earlier, when workspace servers are load balanced, they can be started on the set of servers across which the workspace server is load balanced. These are the steps to starting a workspace server that is load balanced using grid-launched workspace servers:

1. The client application requests the location where SAS Intelligence Platform components are running from the SAS Metadata Server. The SAS Metadata Server returns the server name and connection information to the client.
2. The client sends a request to the SAS Intelligence Platform components running on the server returned from the request to the metadata server for a workspace server.
3. The SAS Intelligence Platform components make a request to SAS Grid Manager to start a workspace server.
4. SAS Grid Manager determines the best server on which to run the next workspace server and starts the workspace server there.
5. The client is assigned the connection to the workspace server that was most recently started.

### 5.01 Poll

You do not need a SAS Metadata Server in a SAS Grid Manager environment.

- True
- False

26

## 5.2 SAS Data Integration Studio with the SAS Grid

---

### Objectives

- Describe SAS Data Integration Studio.
- List the ways that you can exploit the grid within SAS Data Integration Studio.
  - Redirect processing to the grid for single-job sequential execution.
  - Use the loop transform for parallel processing.

29

## SAS Data Integration Studio

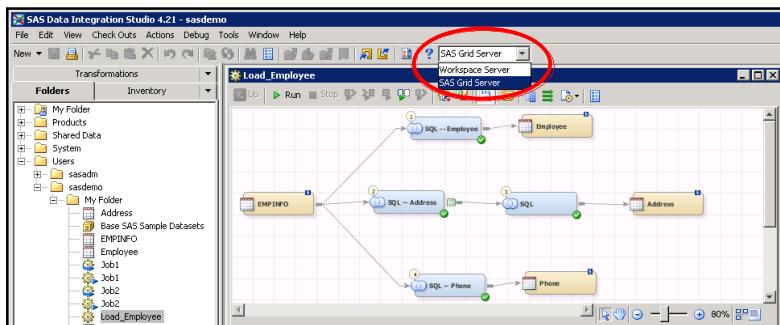
SAS Data Integration Studio has the following characteristics:

- Windows development environment for the construction, execution, and maintenance of data integration projects
- is used to build the following:
  - SAS OLAP Cubes
  - data marts (in SAS data sets or databases)
  - data warehouses (in SAS data sets or databases)
- contains built-in transformations and productivity enhancements

30

Using SAS Data Integration Studio, users build jobs that combine data from various data sources. They clean up the data, add values to the data, summarize the data, and store it in a target data store. Many times, the data stores are either data marts or data warehouses. SAS Data Integration Studio is an interactive development environment with many features to support building these types of jobs. The user creates process flows that are saved as jobs. When finalized, they are scheduled as batch jobs. If you use the Process Manager Scheduler, which is a part of the Platform Suite for SAS (which comes with SAS Grid Manager), your scheduled jobs run on the grid.

## Redirecting to the Grid as One Job



- The entire job is submitted to the SAS Grid Server as one job.
- This spreads work from multiple developers across the grid.
- The job runs in a series and not parallel.

31

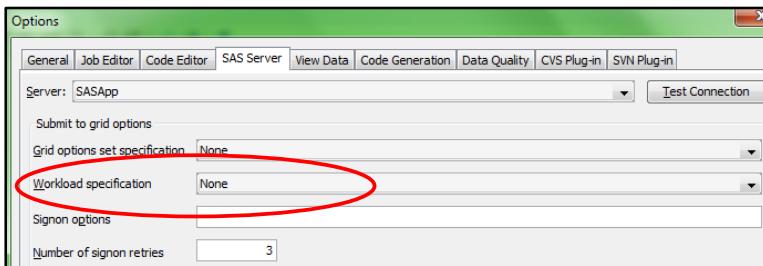
SAS Data Integration Studio enables users to directly submit jobs to the grid. This capability enables the submitted jobs to take advantage of load balancing and the job prioritization that you specified in your grid. SAS Data Integration Studio also enables you to specify the workload that submitted jobs should use. The SAS administrator can then assign servers to process the jobs for the chosen workload.

When a job is run *interactively*, SAS Data Integration Studio jobs can be executed on a grid node. This does not affect **batch** execution in any way. It does not cause parallelization to occur within the job. The three SQL nodes still execute in sequence.

If your SAS Grid Manager environment is configured with load-balanced workspace servers using grid-launched workspace servers, then you should **not** use the SAS Grid Server. When you use grid-launched workspace servers, you are already taking advantage of the grid functionality.

## Selecting Workload Specifications

Designating a workload specification uses Grid Server resource allocations that are configured by your administrator.

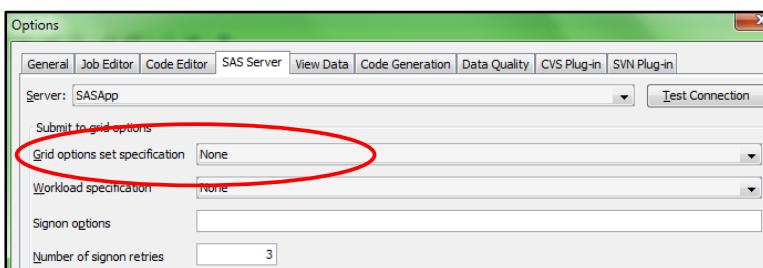


32

The grid *workload specification* is a value that is set up by your SAS administrator. The administrator can use it to map jobs based on the selected workload specification to servers in the grid.

## Selecting a Grid Option Set

You can choose a specific grid option set when your environment is configured to use grid-launched workspace servers.



33

If your SAS Grid Manager environment is load balancing your workspace servers and using grid-launched workspace servers, then you can choose a grid options set to use.

## 5.02 Poll

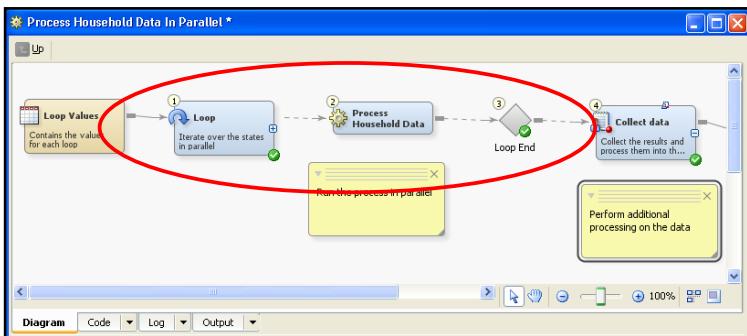
You can run jobs using the workspace server or the grid server from SAS Data Integration Studio.

- True
- False

34

## Loop Transform Parallel Processing

The *loop transform* provides a way to subset a job and run segments of the program in parallel.

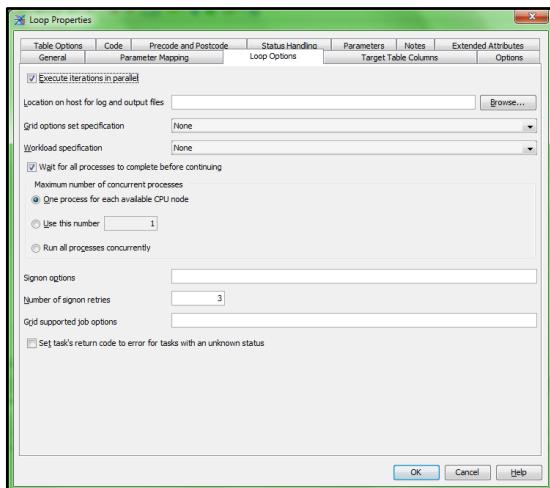


36

In SAS Data Integration Studio, one of the transformations supplied with the product enable you to create a loop to iterate through a set of SAS code based on a set of loop values. Many times, the loop values are pulled from a lookup table and sent to the loop one value at a time. Each iteration through the loop runs the set of code with loop input value.

## Loop Transform Parallel Processing

Properties for the Loop and Loop End transforms can be set so that a job runs in parallel, uses a grid options set, uses a workload specification, and other options.



37

Right-click the **Loop** node. Select **Properties**  $\Rightarrow$  **Loop Options**.

The loop transform has built-in support for running each iteration of the loop in an individual grid session, resulting in parallel execution of several instances of the loop. The number of grid sessions that the loop transform can use can be set by the options. In addition, you can set the grid options set to use and a workload specification to use.

## Loop Transform Parallel Processing

| JobID** | Name          | Status | State Changes | User ID | CPU Usage | CPU Effic |
|---------|---------------|--------|---------------|---------|-----------|-----------|
| 5135    | SASGrid:18508 | DONE   | 1             | \sas    | 0.11s     | 5.47%     |
| 5134    | SASGrid:18508 | DONE   | 1             | \sas    | 0.08s     | 3.91%     |
| 5133    | SASGrid:18508 | DONE   | 1             | \sas    | 0.03s     | 1.56%     |
| 5132    | SASGrid:18508 | DONE   | 1             | \sas    | 0.08s     | 3.91%     |
| 5131    | SASGrid:18508 | DONE   | 1             | \sas    | 0.17s     | 5.73%     |
| 5130    | SASGrid:18508 | DONE   | 1             | \sas    | 0.06s     | 3.13%     |
| 5129    | SASGrid:18508 | DONE   | 1             | \sas    | 0.06s     | 3.13%     |
| 5128    | SASGrid:18508 | DONE   | 1             | \sas    | 0.08s     | 3.91%     |
| 5127    | SASGrid:18508 | DONE   | 1             | \sas    | 0.09s     | 4.69%     |
| 5126    | SASGrid:18508 | DONE   | 1             | \sas    | 0.05s     | 2.34%     |
| 5125    | SASGrid:18508 | DONE   | 1             | \sas    | 0.14s     | 7.03%     |
| 5124    | SASGrid:18508 | DONE   | 1             | \sas    | 0.08s     | 3.91%     |

There is one outer job name and multiple job IDs within the inner loop.

## Steps for Using Loop Transformations

You can set up a parallel processing job in SAS Data Integration Studio in four steps.

- Step 1**  
Create the control table.
  
- Step 2**  
Create the parameterized job.
  
- Step 3**  
Create the iterative job.
  
- Step 4**  
Set loop properties to run in parallel.



## Exercises

---

In this exercise, you use SAS Data Integration Studio to create a data mart table to use for reporting and analysis. You view and run the job both on the workspace server and the grid server.

### 1. Creating a Data Mart Table with SAS Data Integration Studio

- Open SAS Data Integration Studio by selecting **Start** ⇒ **All Programs** ⇒ **SAS** ⇒ **SAS Data Integration Studio**.
- Log on as **Chris** with the password **Student1**.



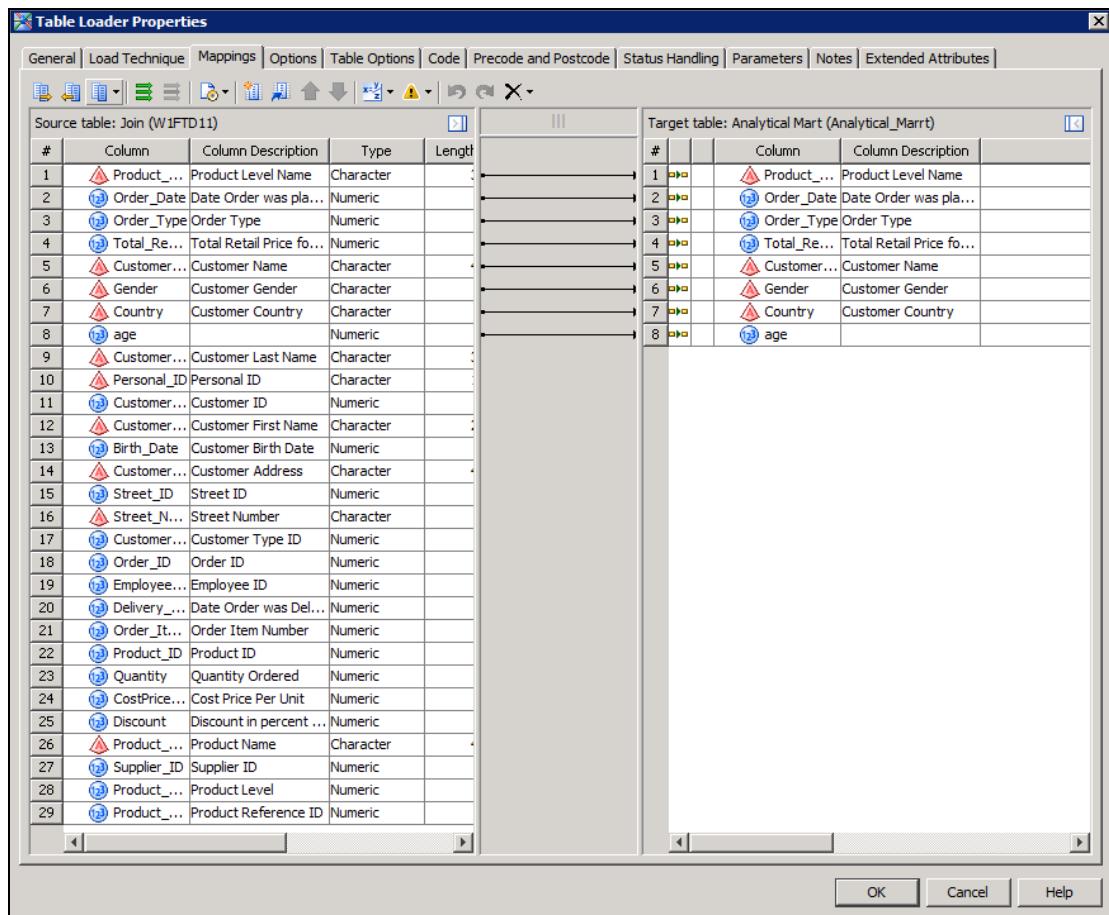
- View the properties of the **Analytical Mart** table.
  - Click the **Folders** tab.
  - Expand the **Orion Inc.** folder.
  - Right-click the **Analytical Mart** table, and select **Properties**.
  - Click the **Columns** tab to view the columns and their properties for the Analytical Mart.
  - Click **Cancel** to return.
- Open the **Load Analytical Mart** job.

While still in the Orion Inc. folder, double-click the **Load Analytical Mart** job.

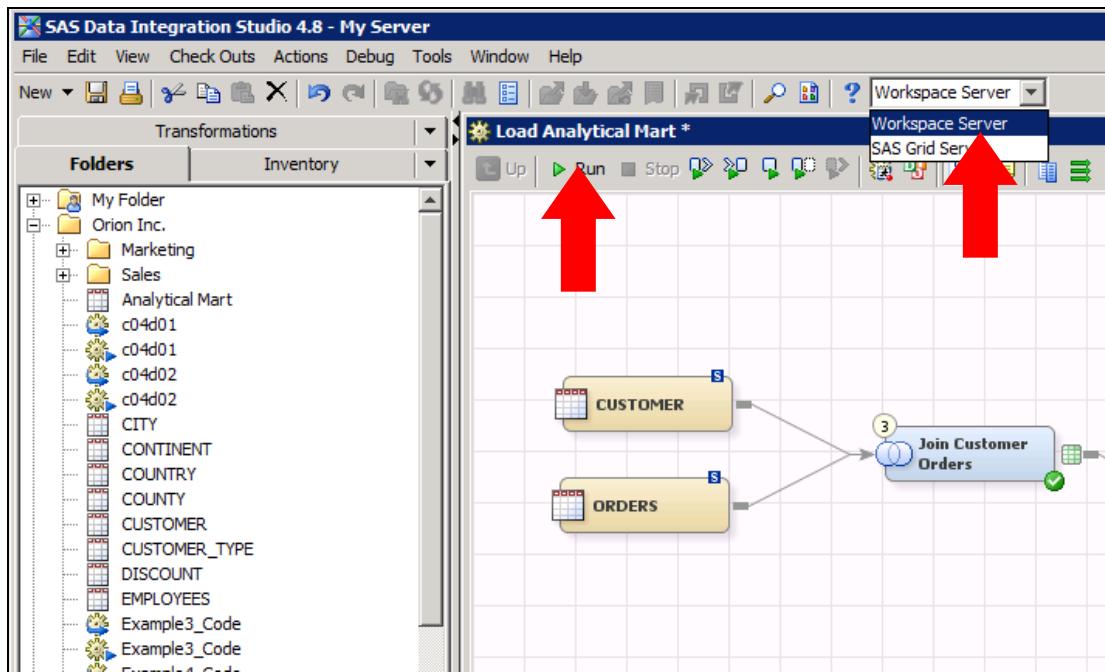
The item with the same name that has a gear icon with a blue arrow on it is the deployed job. This was created when the job was deployed for scheduling.

- Inspect the properties of the join transformations.
  - Double-click each of the join transformations (**Join Customer Orders**, **Product Join**, **Join Product OrderItem**, and **Final Join**) one at a time and perform these steps for each:
    - In the Navigate window on the right, click the **Select** item to see the column mappings for the join.
    - Click the **Where** item to see the join criteria.

- c) Click (up arrow) in the upper left of the join window to return to the top level of the job.
  - d) Repeat these steps for each of the Join transformations in the job.
- 2) Open the **Table Loader Transformation** by double-clicking it.
  - 3) Click the **Mappings** tab to see the final set of variables that are being retained for the Analytical Mart.



- f. Run the Load Analytical Mart job to combine the data from the **Customer**, **Orders**, **Product\_List**, and **Product\_Level** tables.
  - 1) If the Load Analytical Mart job is not open, open it now.
  - 2) Run the job using the workspace server. Select **Workspace Server**  $\Rightarrow$  **Run**.



- 3) Look in one of the monitoring tools (Platform RTM for SAS, Grid Manager Plug-in for SAS Management Console, or the command line utilities) to verify that there are not any grid sessions running while the job runs.

For example, use the command line utilities. On the command line, enter **bjobs -u all**.

You get the following response: No unfinished job found.

- 4) Run the job on the grid. Select **SAS Grid Server**  $\Rightarrow$  **Run**.
- 5) While the job is running, look in one of the monitoring tools to verify that the job is running on a grid server.

In SAS Management Console, click **Job Information** under Grid Monitoring Server to see the job running. (You might need to right-click and select **Refresh**.)

The screenshot shows the SAS Management Console. On the left, a tree view shows 'Foundation', 'Foundation Services Manager', and 'Grid Manager'. Under 'Grid Manager', 'Grid Monitoring Server - sasbap' is expanded, showing 'Job Information', 'Host Information', and 'Queue Information'. A red box highlights the 'Job Information' link. The main pane shows a table of jobs:

| Job ID | Job Name                 | User Name | Status  | Submit Time            | Start Time             |
|--------|--------------------------|-----------|---------|------------------------|------------------------|
| 568    | DIS_Load Analytical Mart | .\Chris   | Running | Jun 2, 2014 1:30:06 PM | Jun 2, 2014 1:30:06 PM |
| 566    | DIS_Load Analytical Mart | .\Chris   | Done    | Jun 2, 2014 1:27:41 PM | Jun 2, 2014 1:27:41 PM |
| 567    | DIS_Load Analytical Mart | .\Chris   | Done    | Jun 2, 2014 1:28:13 PM | Jun 2, 2014 1:28:13 PM |

**End of Exercises**

## 5.3 SAS Enterprise Guide on the SAS Grid

### Objectives

- List SAS Enterprise Guide options to exploit the SAS grid environment.
- Use temporary tables in a grid environment with Enterprise Guide.

42

### Combining SAS Enterprise Guide with SAS Grid Manager

- SAS Enterprise Guide is a GUI application that builds and submits SAS code.
  - provides guided tasks and interfaces for viewing, analyzing, and reporting against data using SAS
  - uses a SAS Workspace Server for executing SAS code that is hand coded or generated by Enterprise Guide
- SAS Grid Manager provides and manages access to a set of servers.
  - determines the best server on which to run code
  - distributes SAS processing among grid servers
- Combining the two provides the advantages of each.
  - a friendly interface to SAS
  - distributed processing across available SAS servers

43

## How Do You Tell Enterprise Guide to Use Grid Sessions?

- SAS Administrator can set server options that **force** Enterprise Guide to use grid sessions.
- You can use the Enterprise Guide options to do the following:
  - start a grid session as soon as the workspace server is started for the user
  - set the properties for a project to use grid sessions
  - set individual items in a process flow to use grid sessions
- ✍ When using grid-launched workspace servers, you should not use the grid options.

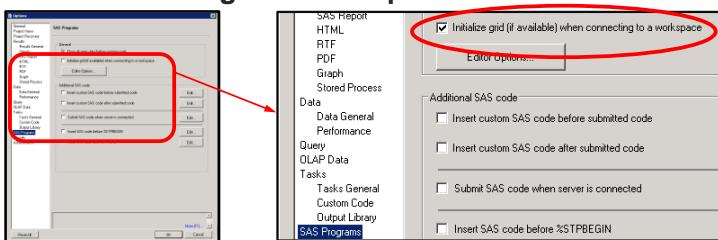
44

Because grid-launched workspace servers use the functionality of the grid to start the workspace server, you should not use the grid in your Enterprise Guide options.

## How Do You Tell Enterprise Guide to Use Grid Sessions?

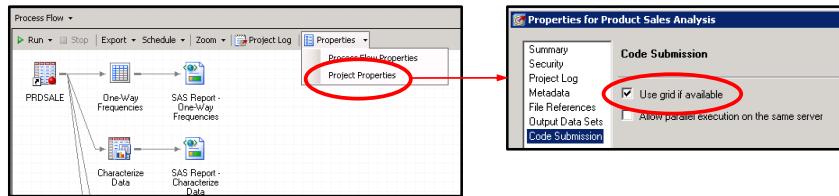
- Start a grid session as soon as the workspace server is started for the user.
- You specify this setting in the Enterprise Guide options.
  - Select **Tools** ⇒ **Options** ⇒ **SAS Programs**.
  - Select the **Initialize grid (if available) when connecting to a workspace** check box.

45



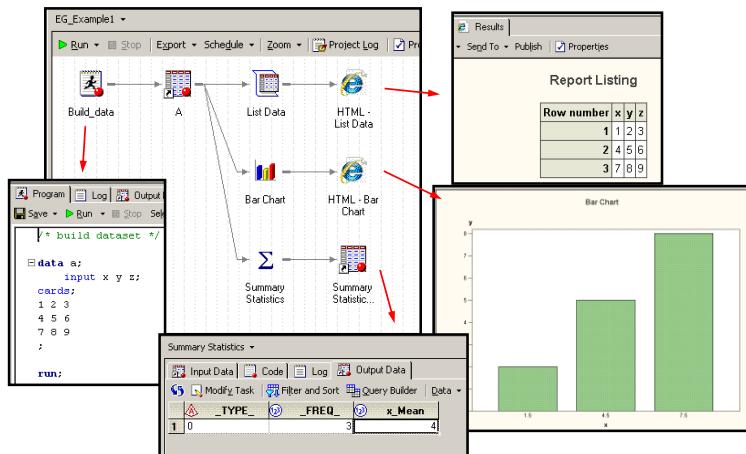
## How Do You Tell Enterprise Guide to Use Grid Sessions?

- Set the properties for a project to use grid sessions.
  - Access the project properties from the following:
    - Select **File** ⇒ **Project Properties**.
    - In the Process Flow window, select **Properties** ⇒ **Project Properties**.



46

## Sample SAS Enterprise Guide Project: Process Flow



47

## Sample SAS Enterprise Guide Project: Running on the Grid

The screenshot shows a SAS Enterprise Guide project titled "EG\_Example1". The project flow consists of three tasks: "Build\_data", "A", and "List Data". Task "A" has two outputs: one to "List Data" and one to a "C:\SAS\GR..." task. The "List Data" task is connected to a "C:\SAS\GR..." task. The log window displays the following text:

```

110
111      rsubmit;
NOTE: Remote submit to TASK1 commencing.
896      data a;
897      input x y z;
898      cards;
NOTE: The data set WORK.A has 3 observations and 3 variables.
NOTE: DATA statement used (Total process time):
      real time          0.01 seconds
      cpu time           0.01 seconds
900      run;
NOTE: Remote submit to TASK1 complete.

```

Red ovals highlight the lines "NOTE: Remote submit to TASK1 commencing." and "NOTE: Remote submit to TASK1 complete.".

48

When you run the SAS Enterprise Guide project on the grid, the only difference is that the RSUBMIT tasks started behind the scenes.

## SAS Management Console: SAS Enterprise Guide on the Grid

The screenshot shows the SAS Management Console interface. On the left, there is a navigation tree under "SAS Management Console" with nodes like "Environment Management", "Grid Manager", and "Metadata Manager". On the right, a table lists two grid jobs:

| # | Job ID            | Job Name | User ... | Status         | Submit T...    | Start Time    | End Time | Exec... | Queue  |
|---|-------------------|----------|----------|----------------|----------------|---------------|----------|---------|--------|
| 1 | 3609 SASGrid:4260 | \\$as    | Running  | Dec 8, 2009... | Dec 8, 2009... |               |          | sashi   | normal |
| 2 | 3078 SASGrid:888  | \\$as    | Done     | Dec 3, 2009... | Dec 3, 2009... | Dec 4, 200... |          | sashi   | normal |

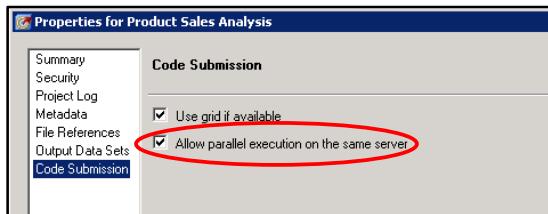
A green callout box points to the table with the text: "Each SAS Enterprise Guide grid session is preserved as a single grid job until it is closed."

49

The grid session starts on the first SAS Enterprise Guide execution and remains active until SAS Enterprise Guide is closed. This enables better iterative work with data, which is typical for a SAS Enterprise Guide user session.

## SAS Enterprise Guide on the Grid

- SAS Enterprise Guide 6.1 gives users the option to run the tasks in a process flow in one session or in multiple sessions.
- This is specified in the Project Properties window.
- Tasks that can run independently run in a separate session.



50

## 5.03 Multiple Choice Poll

Which of the following statements is false?

- a. You can set Enterprise Guide projects to use grid sessions.
- b. You can specify that sections of a task use a grid session.
- c. You can start a grid session with Enterprise Guide when it starts the workspace server.
- d. You can specify that an individual task should use a grid session.

51

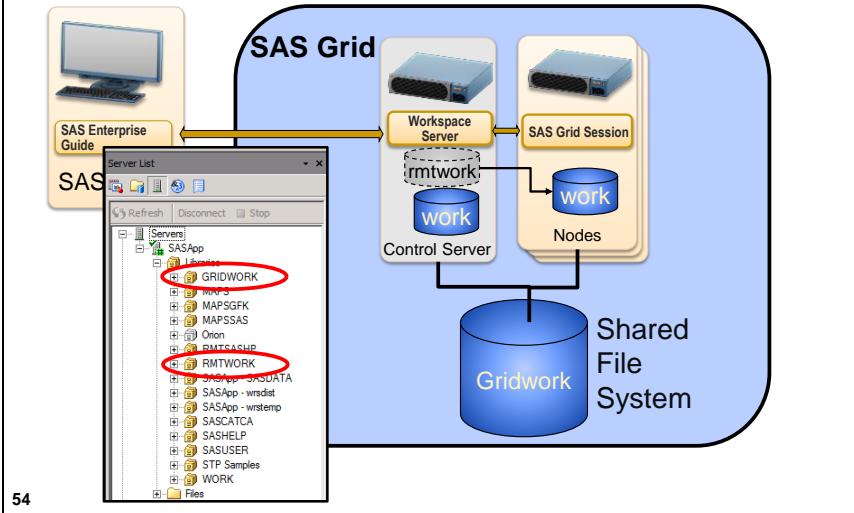
## SAS Enterprise Guide and Temporary Data

Special consideration must be given to using temporary data when using the following:

- a non-load-balanced workspace server
- a load-balanced workspace server that does not use grid-launched workspace servers
- new libraries automatically generated and shown in list of libraries
  - GRIDWORK
  - RMTWORK

53

## Enterprise Guide Work Libraries with Non-Load-Balanced Workspace Servers



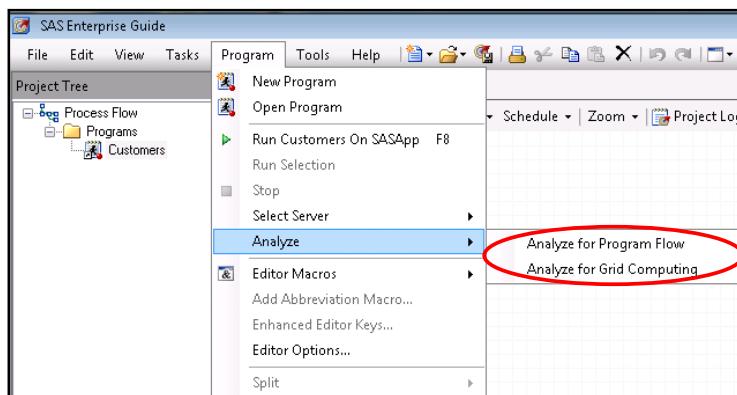
54

## Using the Analyze Program Task

- SAS Enterprise Guide has a task that is based on PROC SCAPROC.
- It provides a way for users to analyze a program for grid computing.

55

## Using the Analyze Program Task



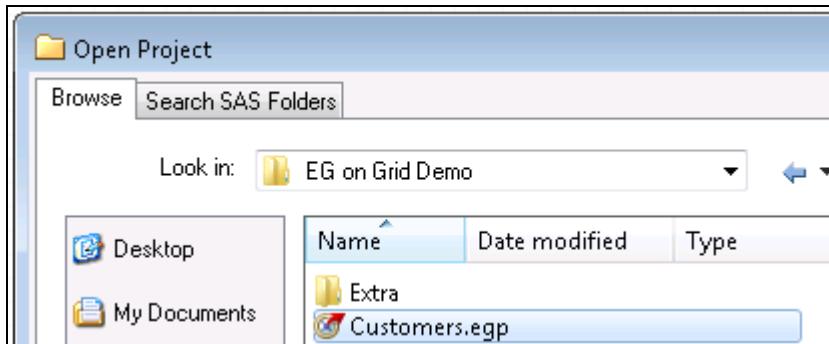
56



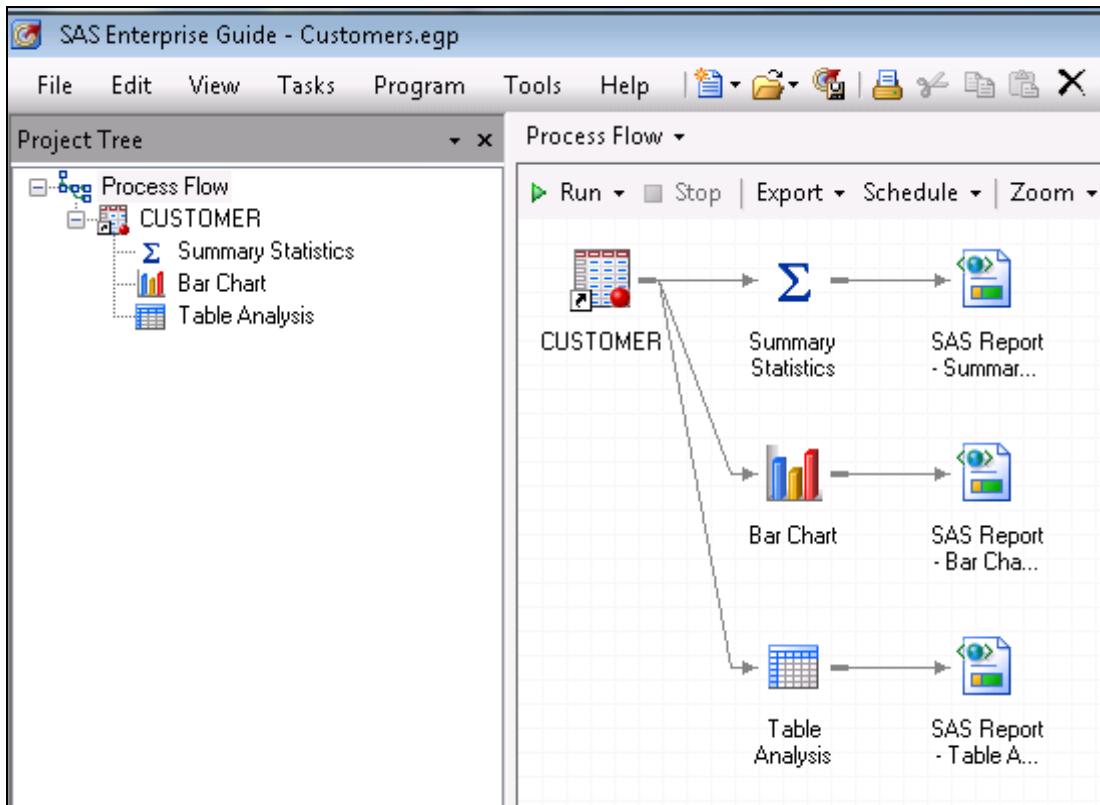
## Using SAS Enterprise Guide on the Grid

1. Open SAS Enterprise Guide by double-clicking the desktop icon.
- a. Under Open a Project, select C:\Grid\Grid Demo Docs\EG on Grid Demo\Customers.egp.  
✎ If the Open a Project list does not list the **Customers.egp** project, select **More Projects**.

Navigate to **C:\Grid\Grid Demo Docs\EG on Grid Demo**. Select the project named **Customers.egp**.



- b. Review the project.

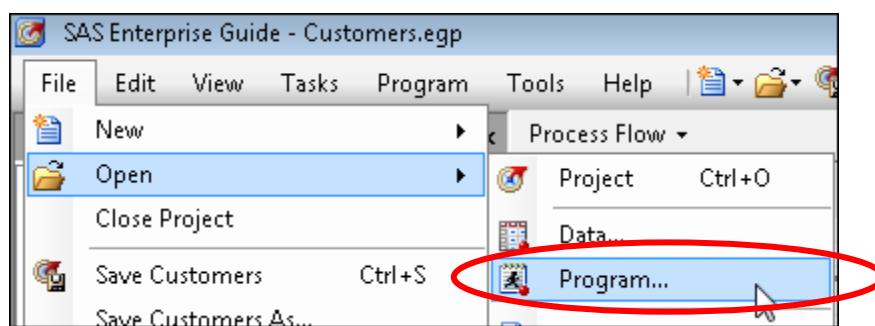


- c. Run the project and notice its status in Platform RTM.

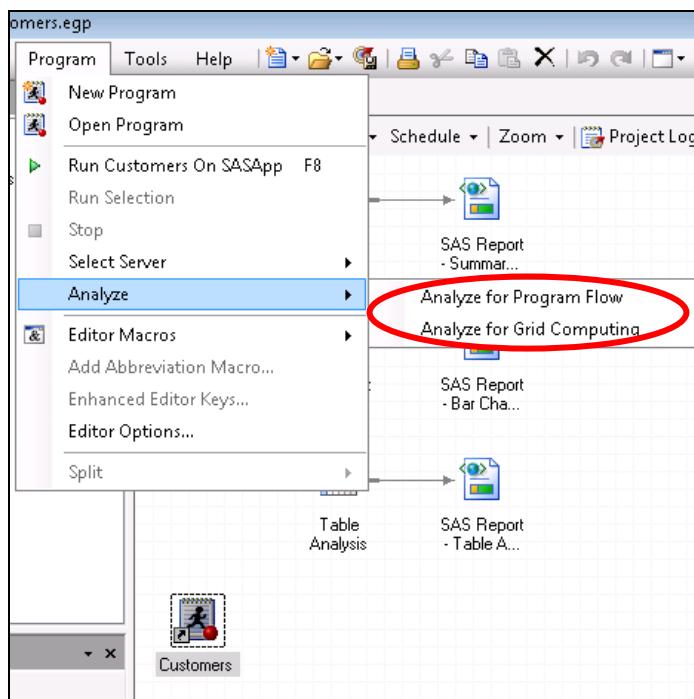
The screenshot shows the 'Batch Job Filters' window with various search criteria set to 'All'. A single job entry is listed:

| Job ID | Name              | Status  | State Changes | User ID  | CPU Usage | CPU Effic | Start Time     |
|--------|-------------------|---------|---------------|----------|-----------|-----------|----------------|
| 2089   | EG.Student.249480 | RUNNING | 2             | \student | -         | -         | 09-23 08:21:32 |

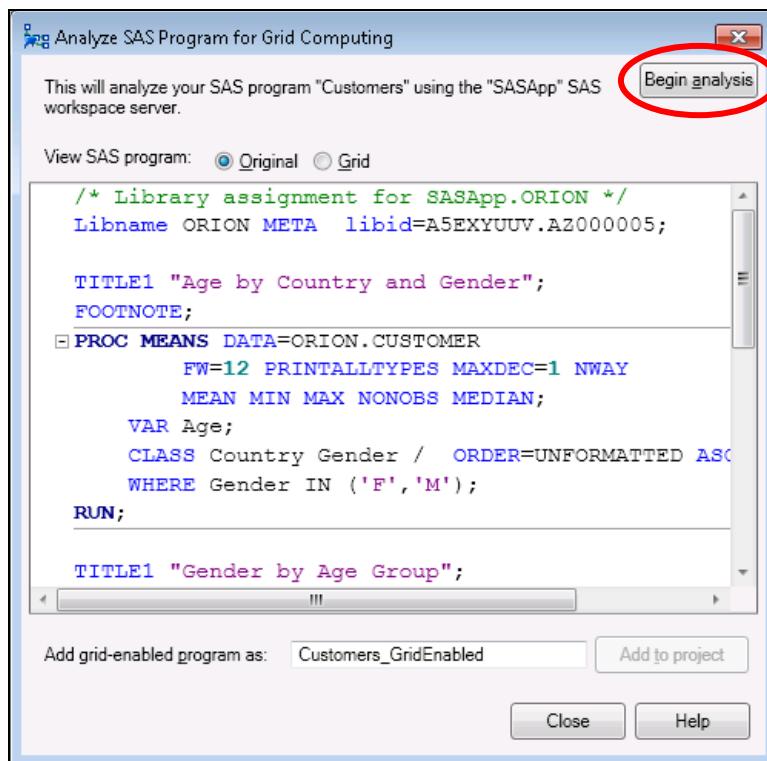
2. Open the program C:\Grid\Grid Demo Docs\EG on Grid Demo\Customers.sas. Select File ⇒ Open ⇒ Program.



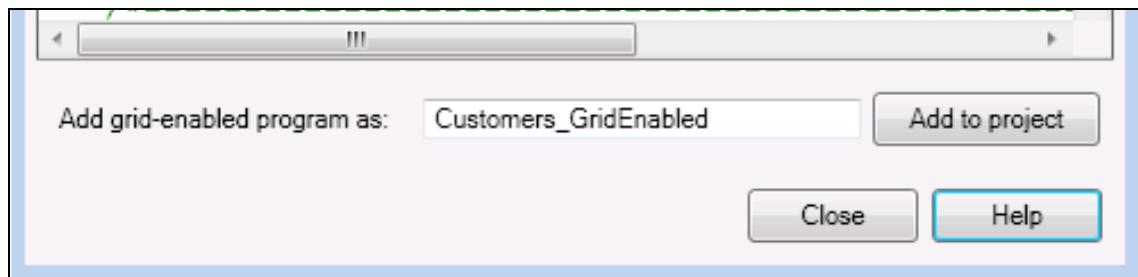
- a. Highlight the program in the process flow. Select Program ⇒ Analyze ⇒ Analyze for Grid Computing.



b. Select **Begin analysis**.



c. When the task completes, select **Add to project**.



d. Review the code.

```

/*
 * There are 2 tasks in this job.
 * 2 of these tasks can be RSUBMITTED.
 * These 2 tasks used 160 units of time.
 * The longest task took 92 units of time, 57.5% of total time.
*/
/*
 * This is the user-modifiable number of connect sessions
 * Numbers of sessions should be between 1 and 20 inclusive.
*/
*let SCAGRID_SESSIONS=3;

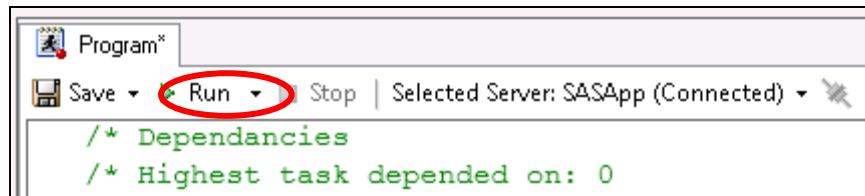
```

- e. Copy the LIBNAME statement from the first task and paste it in the second task.

```
rsubmit &sess sysrputsvnc=yes cmacvar=scagrid_task_2;
Libname ORION META libid=A5EXYUUV.AZ000005;

PROC FREQ DATA = ORION.CUSTOMER
    ORDER=INTERNAL;
    TABLES Age_Group * Gender / NOCUM SCORES=TABLE ALPHA=0.05;
    WHERE Gender NOT = 'G';
RUN;
```

3. Select Run.



4. Review the results.

| Batch Job Filters   |         |               |          |                                             |           |                |               |
|---------------------|---------|---------------|----------|---------------------------------------------|-----------|----------------|---------------|
| Cluster:            | All     | User:         | All      | UGroup:                                     | All       | Status:        | RUNNING       |
| Queue:              | All     | Host:         | All      | HGroup:                                     | All       | Records:       | 30            |
| JobID:              |         | Search:       |          | <input checked="" type="checkbox"/> Dynamic | <b>go</b> | <b>clear</b>   | <b>export</b> |
| << Previous         |         |               |          | Showing Rows 1 to 3 of 3 [1]                |           |                |               |
| JobID** Name        | Status  | State Changes | User ID  | CPU Usage                                   | CPU Effic | Start Time     |               |
| 2101 SASGrid:261476 | RUNNING | 2             | \student | -                                           | -         | 09-23 10:02:23 |               |
| 2100 SASGrid:261476 | RUNNING | 2             | \student | -                                           | -         | 09-23 10:02:13 |               |

#### Partial Output

| The MEANS Procedure     |        |      |         |         |        |
|-------------------------|--------|------|---------|---------|--------|
| Analysis Variable : Age |        |      |         |         |        |
| Country                 | Gender | Mean | Minimum | Maximum | Median |
| AU                      | F      | 42.5 | 19.0    | 74.0    | 43.0   |
|                         | M      | 44.3 | 19.0    | 74.0    | 43.0   |
| BE                      | F      | 46.5 | 19.0    | 74.0    | 48.0   |

5. Close SAS Enterprise Guide. You do not need to save the project.

6. Review the code used to grid-enable SAS Enterprise Guide.

- a. Open Windows Explorer.

- b. Navigate to C:\grid\_work\eg and select the program eg\_amo\_grid.sas.
- c. Right-click and select **TextPad**.

#### Partial Program

```
%macro gridClientPrecode(server=,jobname=,debug=n) ;

%global _GRIDDEBUG;
%local _quotelenopt;

%let _quotelenopt= %sysfunc(getoption(QUOTELENMAX)) ;
options NOQUOTELENMAX;

%gridOneTimeSetup(server=&server,jobname=&jobname,debug=&debug);

%gridDebugBegin;
%gridSetMacroVarsClient;
%gridSetLibsClient;

%if (&_GRIDDEBUG) %then
  %do;
    %put ****;
    %put GRID: Start of global macro variables on client before grid
processing:;
    %put _GLOBAL_;
    %put GRID: End of global macro variables on client before grid
processing.;
    %put ****;
  %end;

  %gridDebugEnd;

  options &_quotelenopt;

%mend;
```

- 7. Close the program.

**End of Demonstration**



## Exercises

---

### 2. Using SAS Enterprise Guide on a SAS Grid

- a. Run a project on the SAS grid.
  - 1) Open the SAS Enterprise Guide project **Employees.egp** located in **C:\Grid\Grid Demo Docs\EG on Grid Demo**.
  - 2) Review the custom code for SAS Enterprise Guide tasks and SAS programs.
  - 3) Run the project and observe the process in Platform RTM.
- b. Open the program **Employees.sas** from **C:\Grid\Grid Demo Docs\EG on Grid Demo**.
  - 1) Analyze the program for SAS Grid Computing.
  - 2) Review the grid-enabled code.
  - 3) Close SAS Enterprise Guide. Select **No** when you are prompted to save the changes.

**End of Exercises**

## 5.4 SAS Add-In for Microsoft Office

### Running a SAS Stored Process on the SAS Grid

---

#### Objectives

- Describe the grid-enablement of tasks in the SAS Add-In for Microsoft Office.

60

#### Redirecting SAS Tasks to the Grid

SAS Add-In for Microsoft Office enables SAS tasks to be submitted to the grid.

To enable the SAS tasks to use the grid, do the following:

- Open a Microsoft Office application.
- Select the SAS Ribbon.
- Select **Tools** ⇒ **Options** ⇒ the **Tasks** tab.
- Select the **Use grid when available** check box.

 When your grid environment is configured to use grid-launched workspace servers, you should **not** select the Use grid when available option check box.

61

## 5.04 Poll

The SAS Add-In for Microsoft Office can run tasks in parallel on the grid.

- True
- False



## Exercises

---

### 3. Using the SAS Add-In for Microsoft Office Running a SAS Stored Process on the SAS Grid

- a. Start Microsoft Excel. Select **Start**  $\Rightarrow$  **All Programs**  $\Rightarrow$  **Microsoft Office 2013**  $\Rightarrow$  **Excel 2013**.
- b. Open a SAS data set to view it in Excel.
  - 1) Click the **SAS** tab to open the SAS Ribbon.
  - 2) Click  **(Open SAS Data)** on the SAS Ribbon.
  - 3) In the View SAS Data window, click **Browse**  $\Rightarrow$  **Analytical Mart**. Use these steps in the Open Data Source window:
    - a) Click **SAS Folders** in the menu pane on the left.
    - b) Drill down into the **Orion Inc.** folder by double-clicking it.
    - c) Click **Analytical Mart**  $\Rightarrow$  **Open**.
    - d) Keep the rest of the defaults. Click **OK**.

Is the workspace server or grid server being used to read the data? \_\_\_\_\_

The data opens in a new workbook.

- 4) Run a SAS task to create a summary table report for **Analytical Mart**.
  - a) Set the SAS options to use the grid.
    - (1) On the SAS Ribbon, click **Tools**  $\Rightarrow$  **Options**.
    - (2) Click the **Results** tab and verify that the **Show status window** and **Show SAS log options** check boxes are selected.
    - (3) Click the **Tasks** tab.
    - (4) Check the **Use Grid when available** check box.
  - b) Click the **Tasks** button on the SAS Ribbon in Excel.
  - c) Select **Describe**  $\Rightarrow$  **Summary Tables Wizard**.
  - d) Select the **External SAS Data** radio button and select **/Orion Inc./Analytical Mart**.
  - e) Under Location for Results, keep the setting for New worksheet. Specify a label for the new worksheet as **Sales of Country by Gender**
  - f) Click **OK** to start the wizard.
  - g) Verify that **Analytical\_Mart** is the selected table and click **Next** to continue.
  - h) On the Analysis Variables page, click **Add**. Click **Total\_Retail\_Price**. Keep **Sum** as the statistic, and click **Next**.

- i) On the Select classification variable page, add **Gender** to the columns and **Country** to the rows. Click **Next**.
- j) Keep the default values for the totals and click **Next**.
- k) Click **Next**. Add a title of **Sales by Gender and Country** and click **Next**.
- l) The task runs and the output is sent to a new worksheet. Two windows appear. One shows the status as the task runs and the other shows the SAS log.

Can you tell whether the task ran on a grid server by looking in the log? If so, how?

---

**End of Exercises**

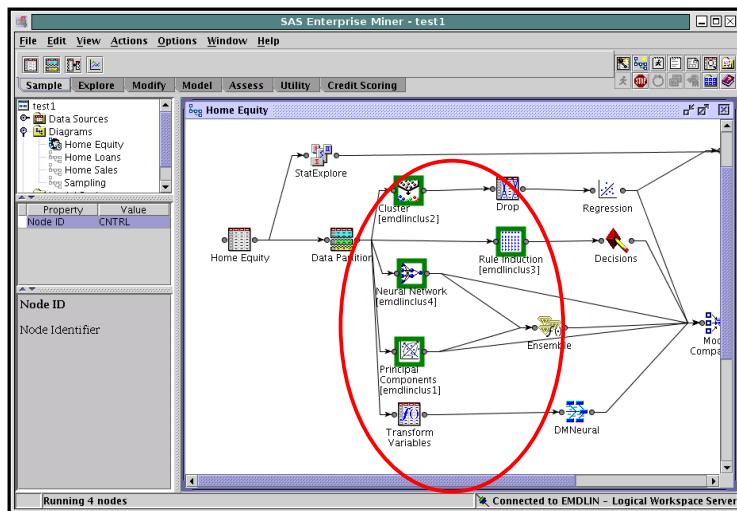
## 5.5 SAS Enterprise Miner on the SAS Grid

### Objectives

- Describe options that are available in SAS Enterprise Miner to enable grid processing.

66

### SAS Enterprise Miner on the Grid



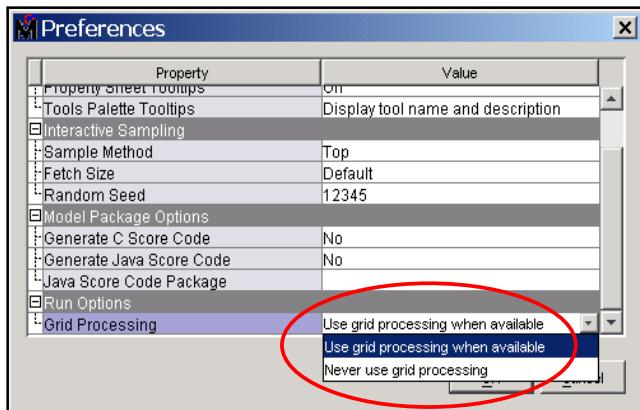
67

Grid enhancements enable SAS Enterprise Miner to leverage parallel node execution in a distributed grid environment. The details of the infrastructure are transparent to the application user. What the user sees is that his or her SAS Enterprise Miner application executes faster. The user only needs to set an option to enable a process to run on the grid. The rest is done automatically.

In addition, even if the SAS Enterprise Miner process is sequential, you get the benefit of workload balancing of multiple SAS Enterprise Miner users.

## SAS Enterprise Miner: Parallelized Workload Balancing

From the Options menu in SAS Enterprise Miner, you can set preferences, including using grid processing.

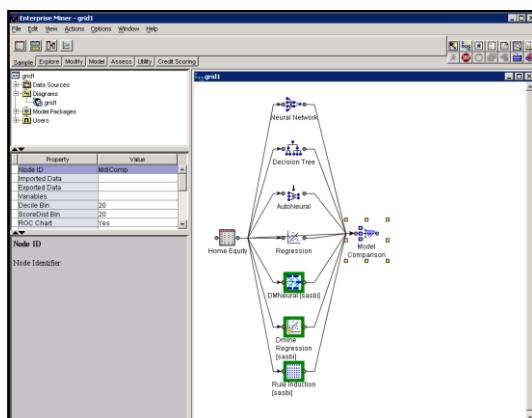


69

Select **Options**  $\Rightarrow$  **Preferences**  $\Rightarrow$  **Use grid processing when available**.

## SAS Enterprise Miner on the Grid Example: Running Parallel Jobs

When grid processing is selected, SAS Enterprise Miner automatically takes advantage of the grid for parallel processing.



70



## Exercises

---

### 4. Using SAS Enterprise Miner on the SAS Grid

- a. Open the SAS Enterprise Miner project named **newtest**.
  - 1) Double-click the **SAS Enterprise Miner** icon on the desktop. Log on with the user ID **Chris** and the password **Student1**.
  - 2) Select **Open Project**. Navigate to **SAS Folders** ⇒ **Users** ⇒ **sasdemo** ⇒ **My Folder** ⇒ **newtest**.
  - 3) Select **Diagrams**. Double-click **Credit\_Grid** to view the SAS Enterprise Miner process flow.
- b. Select the **Use grid processing when available** option. Execute and observe grid tasks in Platform RTM.

**End of Exercises**

# 5.6 Solutions

---

## Solutions to Exercises

In this exercise, you use SAS Data Integration Studio to create a data mart table to use for reporting and analysis. You view and run the job both on the workspace server and the grid server.

### 1. Creating a Data Mart Table with SAS Data Integration Studio

- a. Open SAS Data Integration Studio by selecting **Start**  $\Rightarrow$  **All Programs**  $\Rightarrow$  **SAS**  $\Rightarrow$  **SAS Data Integration Studio**.
- b. Log on as **Chris** with the password **Student1**.



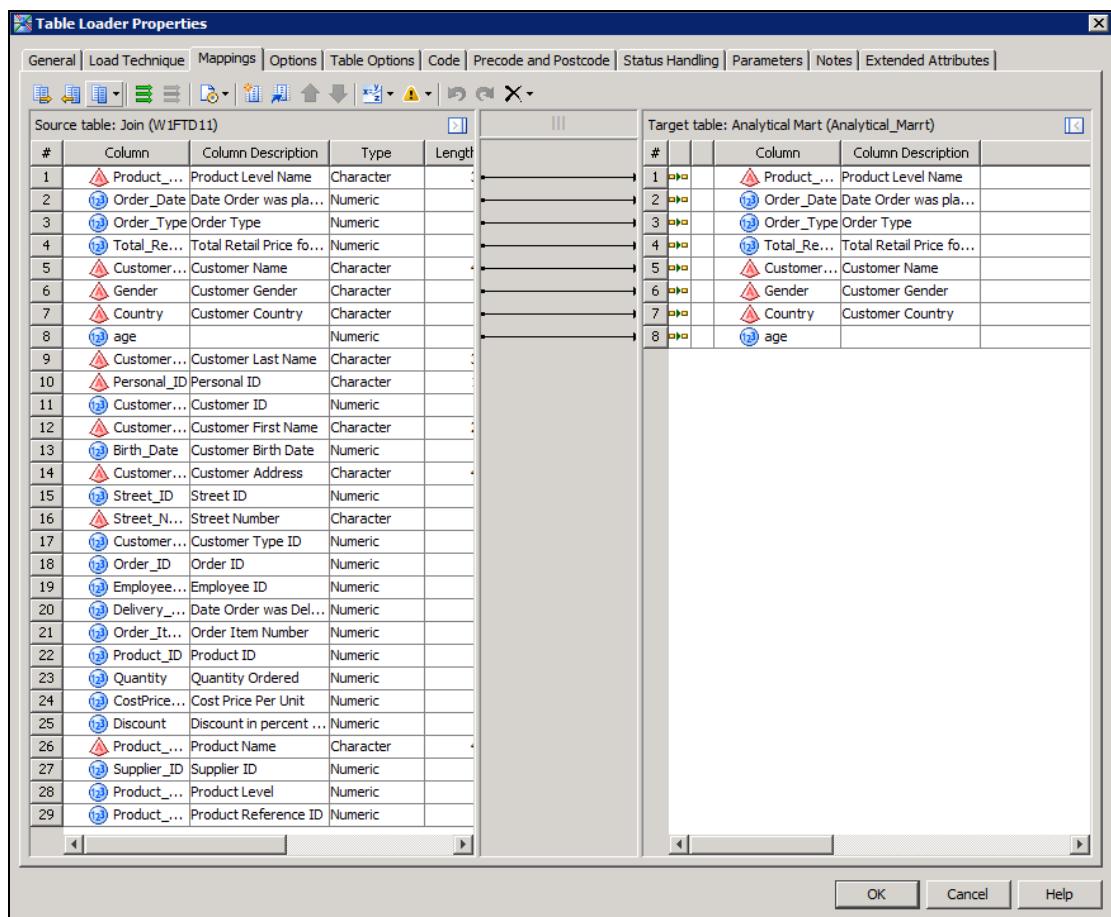
- c. View the properties of the **Analytical Mart** table.
  - 1) Click the **Folders** tab.
  - 2) Expand the **Orion Inc.** folder.
  - 3) Right-click the **Analytical Mart** table and select **Properties**.
  - 4) Click the **Columns** tab to view the columns and their properties for the Analytical Mart.
  - 5) Click **Cancel** to return.
- d. Open the **Load Analytical Mart** job.

While still in the Orion Inc. folder, double-click the **Load Analytical Mart** job.

The item with the same name that has a gear icon with a blue arrow on it is the deployed job. This was created when the job was deployed for scheduling.

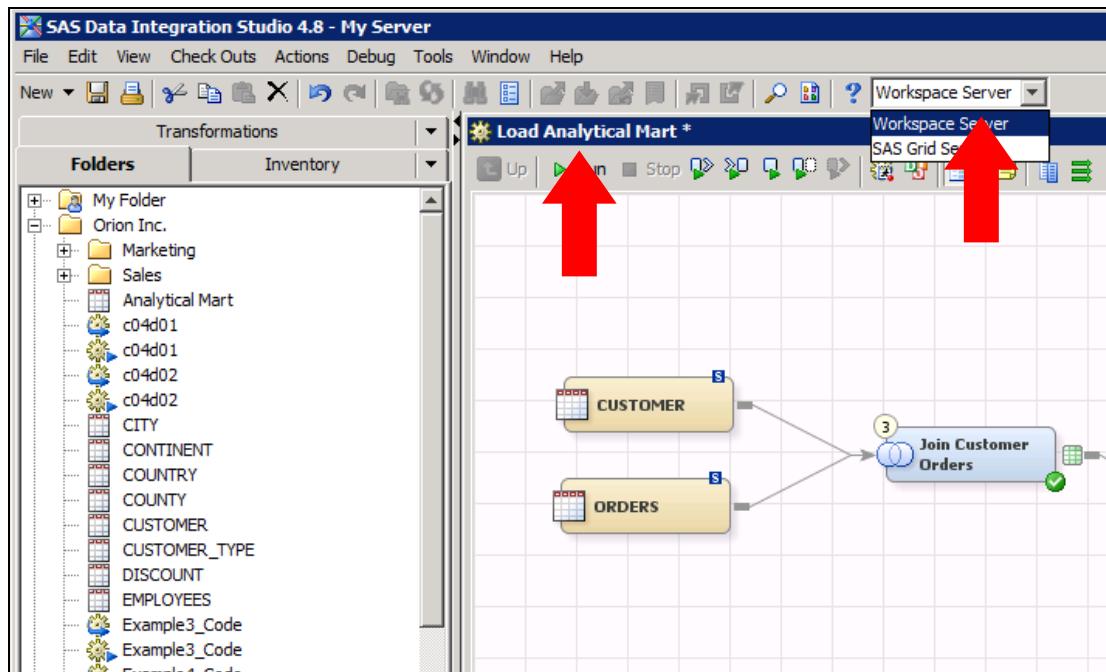
- e. Inspect the properties of the join transformations.
  - 1) Double-click each of the join transformations (**Join Customer Orders**, **Product Join**, **Join Product OrderItem**, and **Final Join**) one at a time and perform these steps for each:
    - a) In the Navigate window on the right, click the **Select** item to see the column mappings for the join.
    - b) Click the **Where** item to see the join criteria.

- c) Click (up arrow) in the upper left of the join window to return to the top level of the job.
  - d) Repeat these steps for each of the Join transformations in the job.
- 2) Open the **Table Loader Transformation** by double-clicking it.
  - 3) Click the **Mappings** tab to see the final set of variables that are being retained for the Analytical Mart.



- f. Run the Load Analytical Mart job to combine the data from the **Customer**, **Orders**, **Product\_List**, and **Product\_Level** tables.
- 1) If the Load Analytical Mart job is not open, open it now.

- 2) Run the job using the workspace server. Select **Workspace Server**  $\Rightarrow$  **Run**.



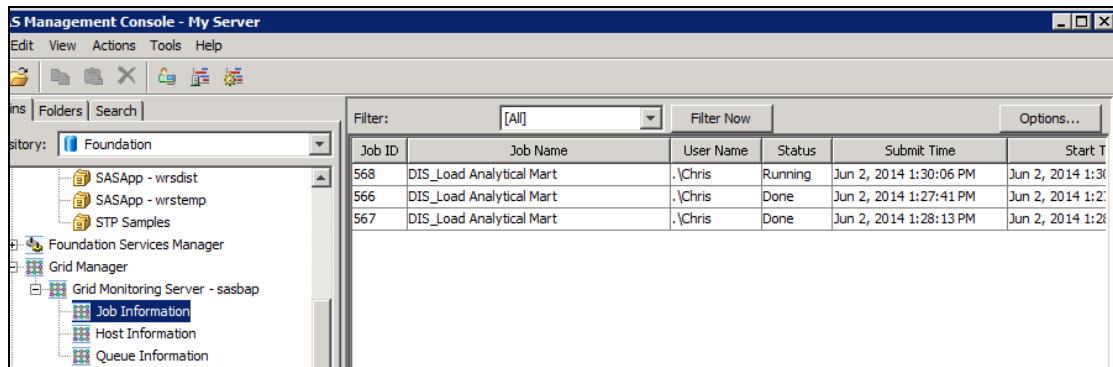
- 3) Look in one of the monitoring tools (Platform RTM for SAS, Grid Manager Plug-in for SAS Management Console, or the command line utilities) to verify that there are not any grid sessions running while the job runs.

For example, use the command line utilities. On the command line, enter **bjobs -u all**.

You get the following response: No unfinished job found.

- 4) Run the job on the grid. Select **SAS Grid Server**  $\Rightarrow$  **Run**.  
 5) While the job is running, look in one of the monitoring tools to verify that the job is running on a grid server.

In SAS Management Console, click **Job Information** under Grid Monitoring Server to see the job running. (You might need to right-click and select **Refresh**.)

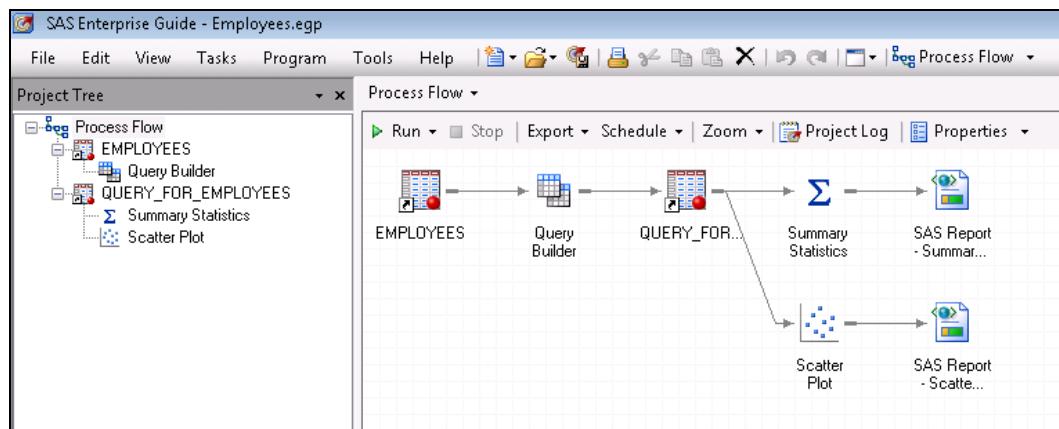


## 2. Using SAS Enterprise Guide on a SAS Grid

### a. Run a project on the SAS grid.

- 1) Open the SAS Enterprise Guide project **Employees.egp** located in **C:\Grid\Grid Demo Docs\EG on Grid Demo**.

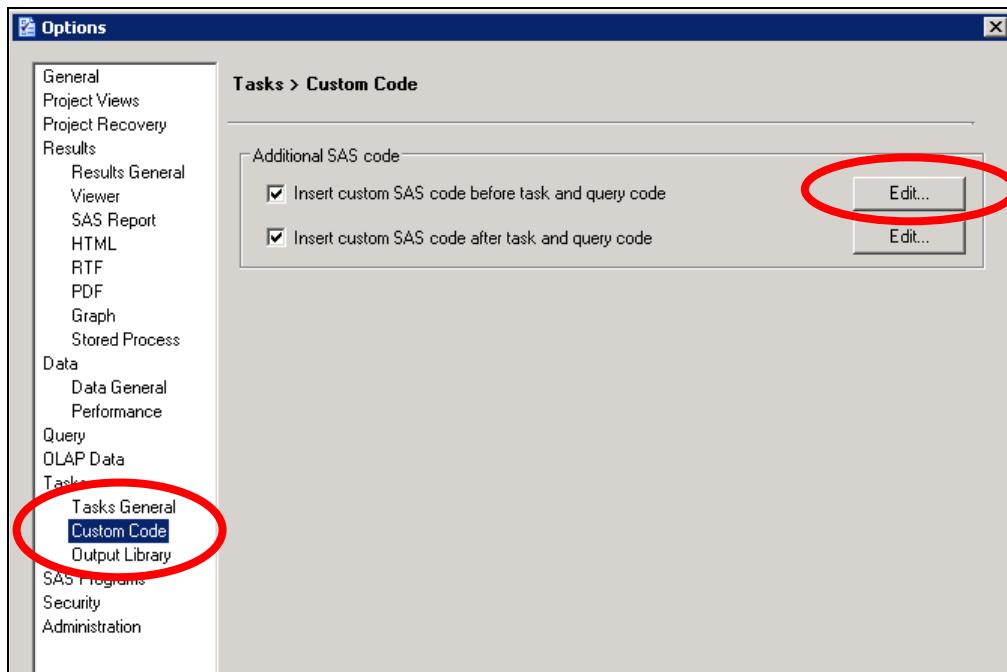
  - a) Double-click the **SAS Enterprise Guide** icon on the desktop. SAS Enterprise Guide opens.
  - b) Under Open a Project, select **C:\Grid\Grid Demo Docs\EG on Grid Demo\Employees.egp**.
    - If the Open a Project list does not list the **Customers.egp** project, select **More Projects**. Navigate to **C:\Grid\Grid Demo Docs\EG on Grid Demo** and select the project named **Employees.egp**.
    - If SAS Enterprise Guide is already open, select **File**  $\Rightarrow$  **Open**  $\Rightarrow$  **Project** and navigate to **C:\Grid\Grid Demo Docs\EG on Grid Demo**. Select **Employees.egp**.



- 2) Review the custom code for SAS Enterprise Guide tasks and SAS programs.

- a) Select **Tools**  $\Rightarrow$  **Options**.
  - (1) In the left pane, select **Tasks**  $\Rightarrow$  **Custom Code**.

- (2) Select **Edit** next to Insert custom SAS code before task and query code.



- (3) Review the code.

Partial Code

```
%gridClientPrecode;
rsubmit grid &_GridInheritLibs;
%gridRemotePrecode;
```

- (4) Select **Cancel**.

- b) Repeat the process for the post-code.

- (1) Review the code.

- (2) Select **Cancel**.

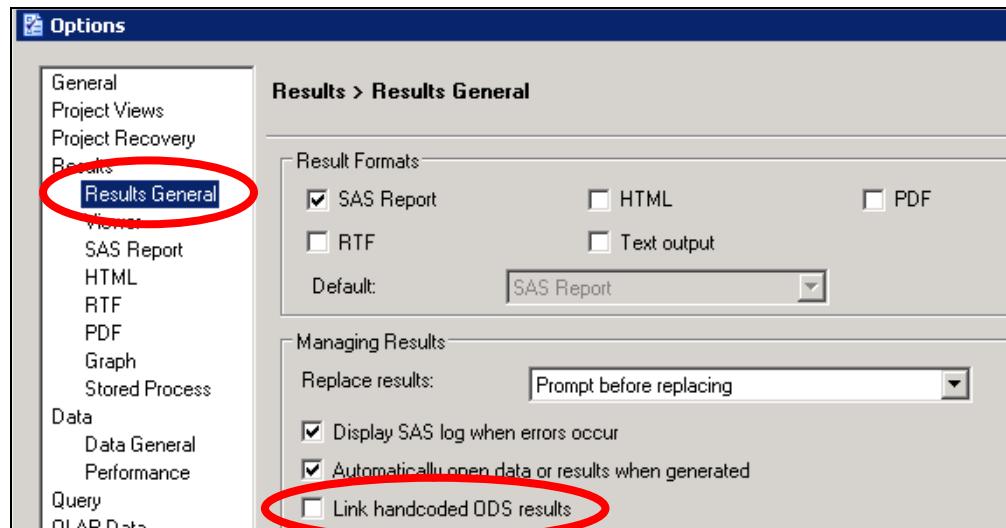
- c) Repeat the process for SAS programs.

- (1) Review the pre- and post-code (which is identical to that for tasks).

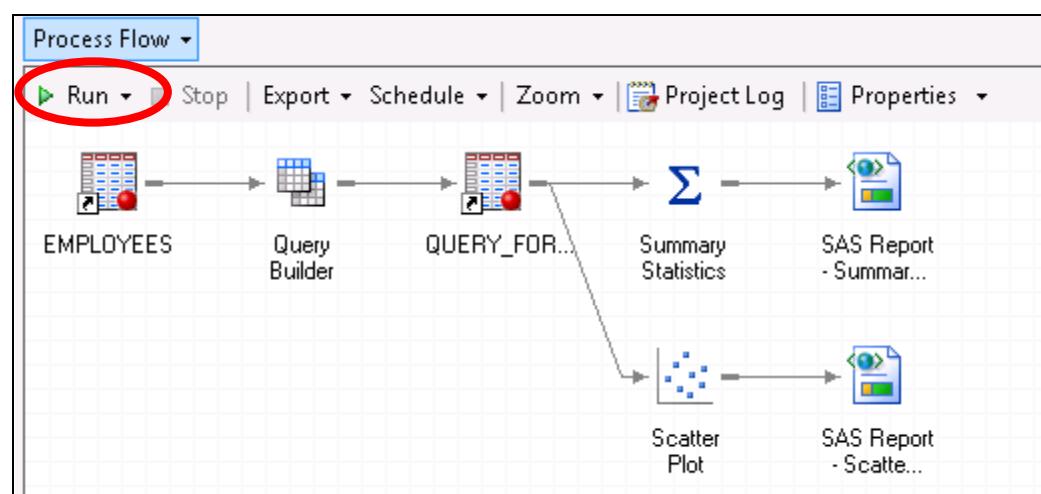
- (2) Select **Cancel**.

- d) Select **Results**  $\Rightarrow$  **Results General**.

- (1) Notice that the **Link handcoded ODS results** check box is clear.



- (2) Select **Cancel**.
- 3) Run the project and observe the process in Platform RTM.
- a) Click **Run** ⇒ **Run Project** above the process flow.



- b) Go to Platform RTM to observe the results.

| Batch Job Filters      |         |               |          |           |           |                              |               |
|------------------------|---------|---------------|----------|-----------|-----------|------------------------------|---------------|
| Cluster:               | All     | User:         | All      | UGroup:   | All       | status:                      | ACTIVE        |
| Queue:                 | All     | Host:         | All      | HGroup:   | All       | Records:                     | 30            |
| JobID:                 |         | Search:       |          | Dynamic   | <b>go</b> | <b>clear</b>                 | <b>export</b> |
| << Previous            |         |               |          |           |           | Showing Rows 1 to 1 of 1 [1] |               |
| Job Name               | Status  | State Changes | User ID  | CPU Usage | CPU Effic | Start Time                   |               |
| 2090 EG.Student.260588 | RUNNING | 2             | \student | -         | -         | 09-23 09:23:40               |               |

- c) Close SAS Enterprise Guide.

- d) Select **No** when you are prompted to save the changes.
- b. Open the program **Employees.sas** from C:\Grid\Grid Demo Docs\EG on Grid Demo.
- 1) Analyze the program for SAS Grid Computing.

- a) Select **File**  $\Rightarrow$  **Open**  $\Rightarrow$  **Program** and select **Employees.sas**.

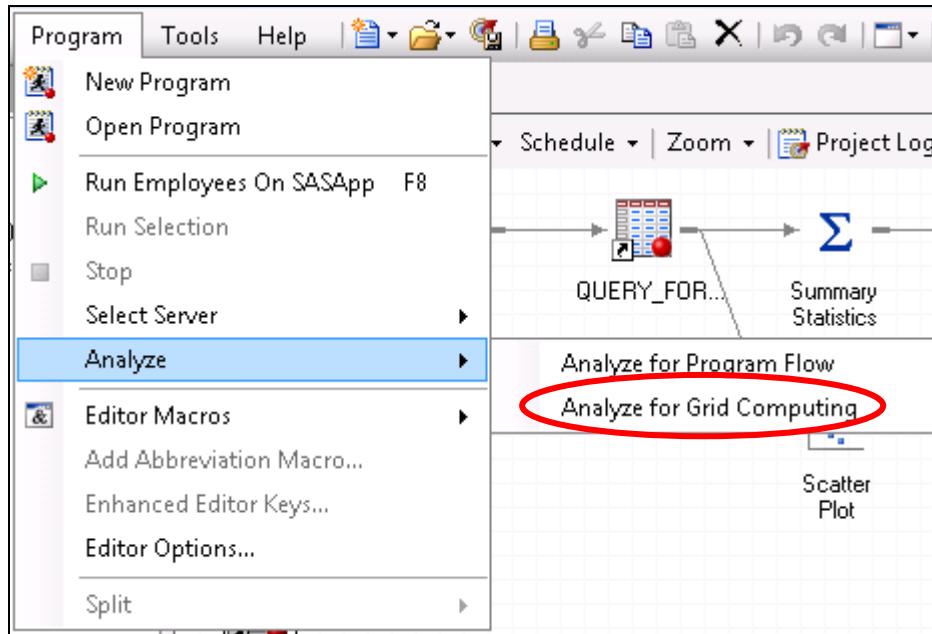
```
/* Employees.sas */

title1 "Salary by Department";
libname orion meta libid=a5exyuuv.az000005;
proc means data= orion.employees
            fw=12 printalltypes chartype
            maxdec=2 nway mean min max n;
    var salary;
    class department / order=unformatted ascending;
    where termdate is missing;
run;

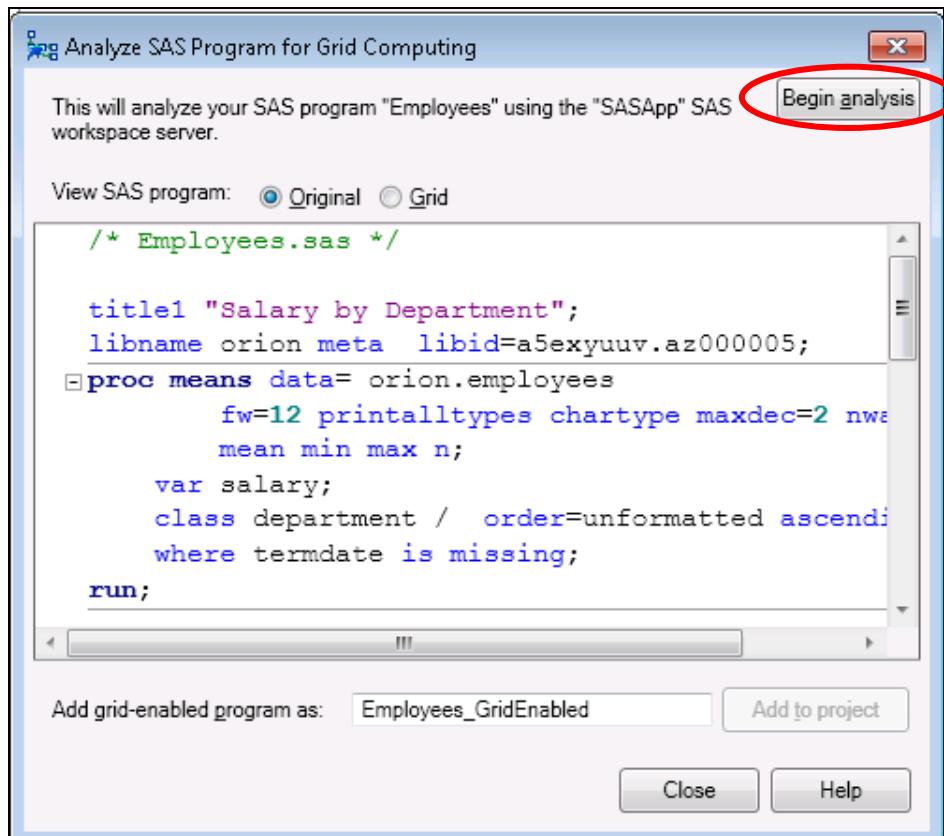
title 'Count of Employees by Country and Starting Year';
libname orion meta libid=a5exyuuv.az000005;
proc freq data=orion.employees;
    tables country*hiredate / nocol;
    format hiredate year4.;
run;

title 'Count of Employees by Country and Starting Year';
libname orion meta libid=a5exyuuv.az000005;
proc corr data=orion.employees nosimple;
    var salary hiredate;
    format hiredate year4.;
run;
```

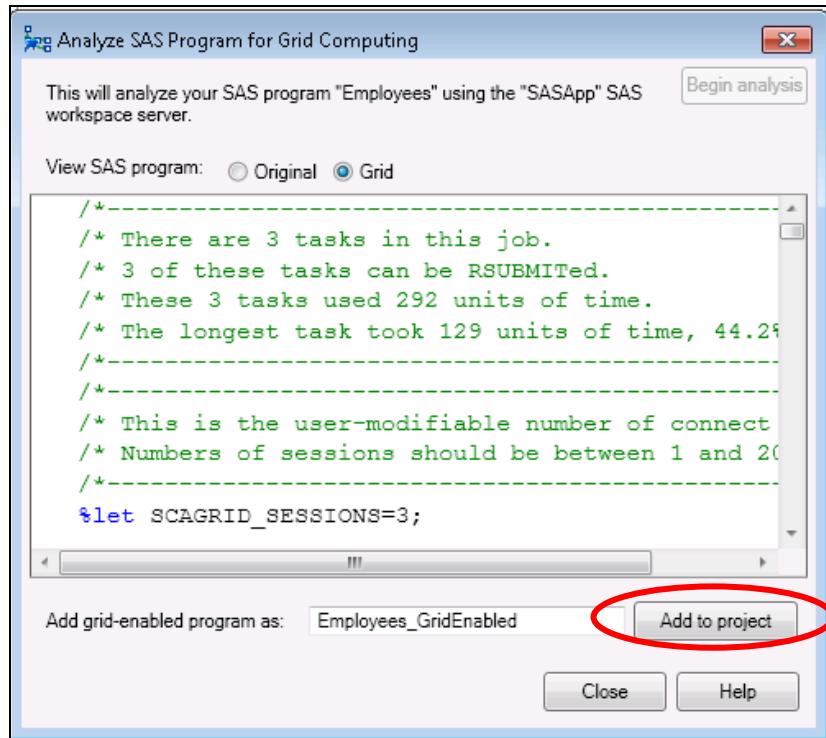
- b) Highlight the program and select **Programs** ⇒ **Analyze** ⇒ **Analyze for Grid Computing**.



- c) Select **Begin Analysis**.



- d) When the analysis completes, select **Add to Project**.



- 2) Review the grid-enabled code.

#### Partial Program

```
/*
/* There are 3 tasks in this job. */
/* 2 of these tasks can be RSUBMITTED. */
/* These 3 tasks used 292 units of time. */
/* The longest task took 129 units of time, 44.2% of total time. */
/*
/* This is the user-modifiable number of connect sessions */
/* Numbers of sessions should be between 1 and 20 inclusive. */
/*
%let SCAGRID_SESSIONS=3;

/*
/* Enable grid service */
/*
%let rc=%sysfunc(grdsvc_enable(_all_, resource=SASApp));
```

- a) To run the program, you might need to move or copy the LIBNAME statement into the last two RSUBMIT blocks. The final code should include what is shown below.

## Partial Program

```
rsubmit &sess sysrputsync=yes cmacvar=scagrid_task_2;
libname orion meta libid=a5exyuuv.az000005;
title 'Count of Employees by Country and Starting Year';
proc freq data=orion.employees;
  tables country*hiredate / nocol;
  format hiredate year4.;
run;

endrsubmit;

...
rsubmit &sess sysrputsync=yes cmacvar=scagrid_task_3;
libname orion meta libid=a5exyuuv.az000005;
proc corr data=orion.employees nosimple;
  var salary hiredate;
  format hiredate year4.;
run;
```

- b) If you run the program, review the processing in Platform RTM and view the results.

| JobID | Name              | Status  | State Changes | User ID  | CPU Usage | CPU Effic | Start Time     |
|-------|-------------------|---------|---------------|----------|-----------|-----------|----------------|
| 2103  | EG.Student.269808 | RUNNING | 2             | \student | -         | -         | 09-23 10:25:05 |
| 2104  | SASGrid:274900    | RUNNING | 2             | \student | -         | -         | 09-23 10:36:27 |

## Partial Output

| The FREQ Procedure           |                     |                   |                   |                   |                   |                   |                   |                   |                   |
|------------------------------|---------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Table of Country by HireDate |                     |                   |                   |                   |                   |                   |                   |                   |                   |
| Country                      | HireDate            |                   |                   |                   |                   |                   |                   |                   |                   |
| Frequency                    |                     |                   |                   |                   |                   |                   |                   |                   |                   |
| Percent                      |                     |                   |                   |                   |                   |                   |                   |                   |                   |
| Row Pct                      | 1974                | 1975              | 1976              | 1977              | 1978              | 1979              | 1980              | 1981              | 1982              |
| AU                           | 20<br>1.91<br>18.52 | 0<br>0.00<br>0.00 | 1<br>0.10<br>0.93 | 1<br>0.10<br>0.93 | 4<br>0.38<br>3.70 | 4<br>0.38<br>3.70 | 2<br>0.19<br>1.85 | 1<br>0.10<br>0.93 | 4<br>0.38<br>3.70 |
| BE                           | 0<br>0.00<br>0.00   | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 | 0<br>0.00<br>0.00 |

- 3) Close SAS Enterprise Guide. Select No when you are prompted to save the changes.

### 3. Using the SAS Add-In for Microsoft Office to Run a SAS Task on the SAS Grid

- a. Start Microsoft Excel using Start ⇒ All Programs ⇒ Microsoft Office 2013 ⇒ Excel 2013

**b.** Open a SAS Data Set to view it in Excel

- 1) Click the **SAS** tab to open the SAS Ribbon.



- 2) Click the **Open SAS Data** icon on the SAS Ribbon:
- 3) In the View SAS Data Window, click **Browse** and select **Analytical Mart** using these steps in the Open Data Source window:
  - a) Click **SAS Folders** in the menu pane on the left
  - b) Drill down into the **Orion Inc.** folder by double-clicking it.
  - c) Click **Analytical Mart**  $\Rightarrow$  **Open**.
  - d) Keep the rest of the defaults and click **OK**.
    - (1) Is the workspace server or grid server being used to read the data?

**The workspace server is used to read the data. By clicking on the jobs in the Grid Manager plug-in, you see that not any grid jobs are running.**

- 4) The data opens in a new workbook.
- 5) Run a SAS task to create a summary table report for **Analytical Mart**.
  - a) Set the SAS options to use the grid.
    - (1) On the SAS Ribbon, click **Tools**  $\Rightarrow$  **Options**.
    - (2) Click the **Results** tab and verify that the **Show status window** and **Show SAS log options** are selected.
    - (3) Click the **Tasks** tab.
    - (4) Check the box for **Use Grid when available**.
  - b) Click the **Tasks** button on the SAS Ribbon in Excel.
  - c) Select **Describe**  $\Rightarrow$  **Summary Tables Wizard**.
  - d) Click the radio button for **External SAS Data** and select **/Orion Inc./Analytical Mart**.
  - e) Under Location for Results keep the setting for New worksheet. Specify a label for the new worksheet as **Sales of Country by Gender**.
  - f) Click **OK** to start the wizard.
  - g) Verify that **Analytical\_Mart** is the selected table and click **Next** to continue.
  - h) On the Analysis Variables page, click **Add**. Then click **Total\_Retail\_Price**. Keep **Sum** as the statistic. Click **Next**.
  - i) In the Select classification variable page, add **Gender** to the columns, and **Country** to the rows. Click **Next**.

- j) Keep the default values for the totals and click **Next**.
- k) Click **Next**. Add a title of **Sales by Gender and Country** and click **Next**.
- l) The task runs and the output is sent to a new worksheet and two windows appear. One shows the status as the task runs and another shows the SAS log.
- m) Can you tell whether the task ran on a grid server by looking in the log? If so, how?

The following statements from the SAS log show that indeed the task did execute on the grid node SASBAP:

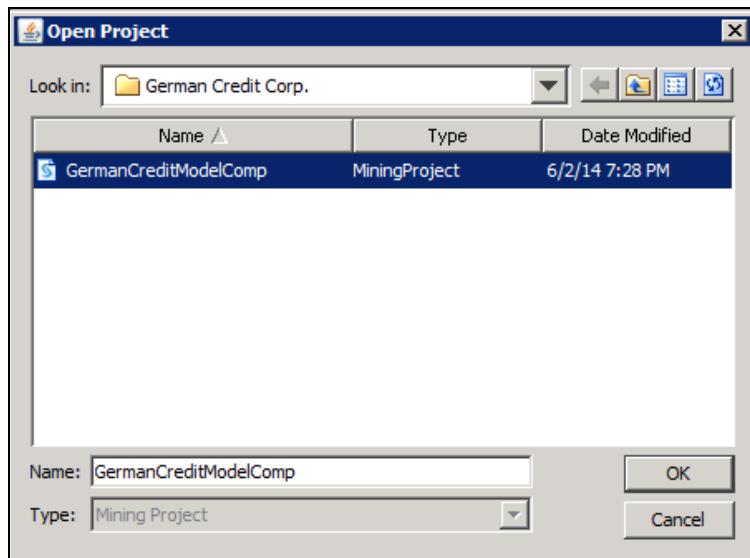
```
NOTE: Remote submit to GRID commencing.  
1492 %put NOTE: Using grid node &SYSHOSTNAME to run  
task.;  
  
NOTE: Using grid node SASBAP to run task.
```

#### 4. Using SAS Enterprise Miner on the SAS Grid

- a. Open the SAS Enterprise Miner project named **GermanCreditModelComp**.
- 1) Double-click the **SAS Enterprise Miner** icon on the desktop. Log on with the user ID **Bryan** and the password **Student1**.
- 2) Select **Open Project**.

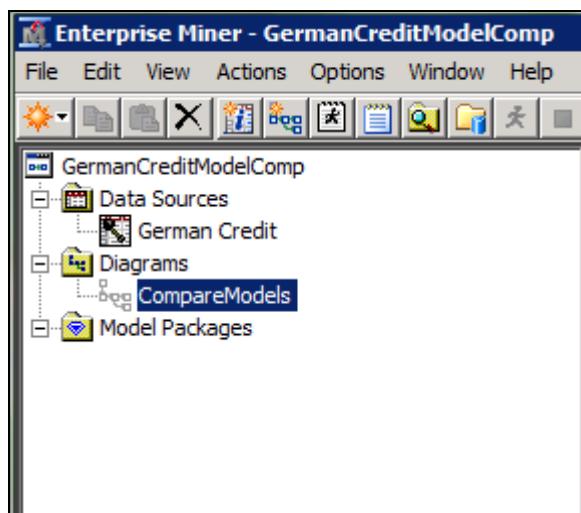


- a) Select **SAS Folders** ⇒ **German Credit Corp.**

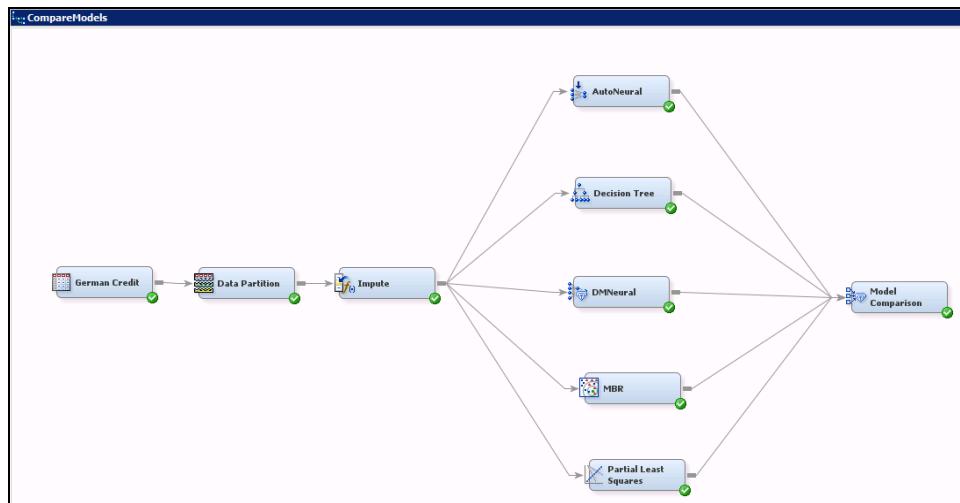


- b) Select **GermanCreditModelComp**.  
c) Click **OK**.
- SAS Enterprise Miner appears.  
3) Select **Diagrams**.

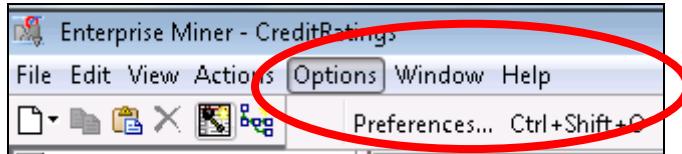
- a) Click the **plus sign** to expand it.



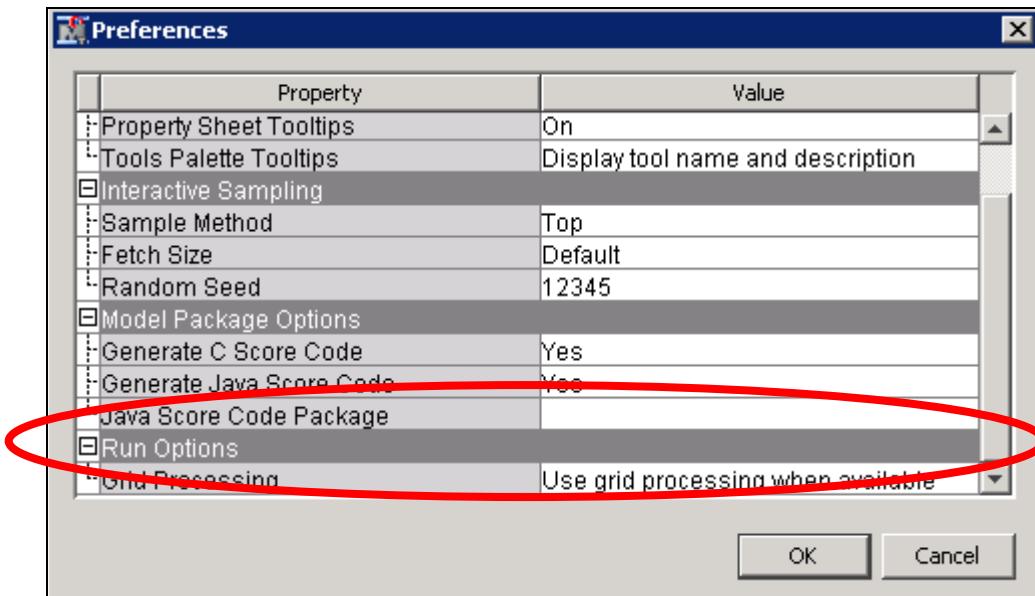
- b) Double-click **CompareModels** to view the SAS Enterprise Miner process flow.



- b. Select the **Use grid processing when available** option. Execute and observe grid tasks in Platform RTM.
- 1) Select **Options**  $\Rightarrow$  **Preferences** on the menu.

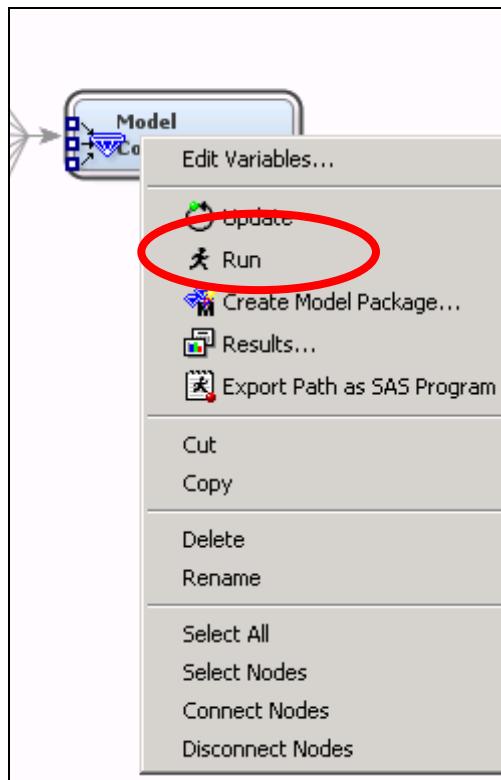


- 2) Scroll to the bottom of the list of options and verify that the grid processing option is set to **Use grid processing when available**.

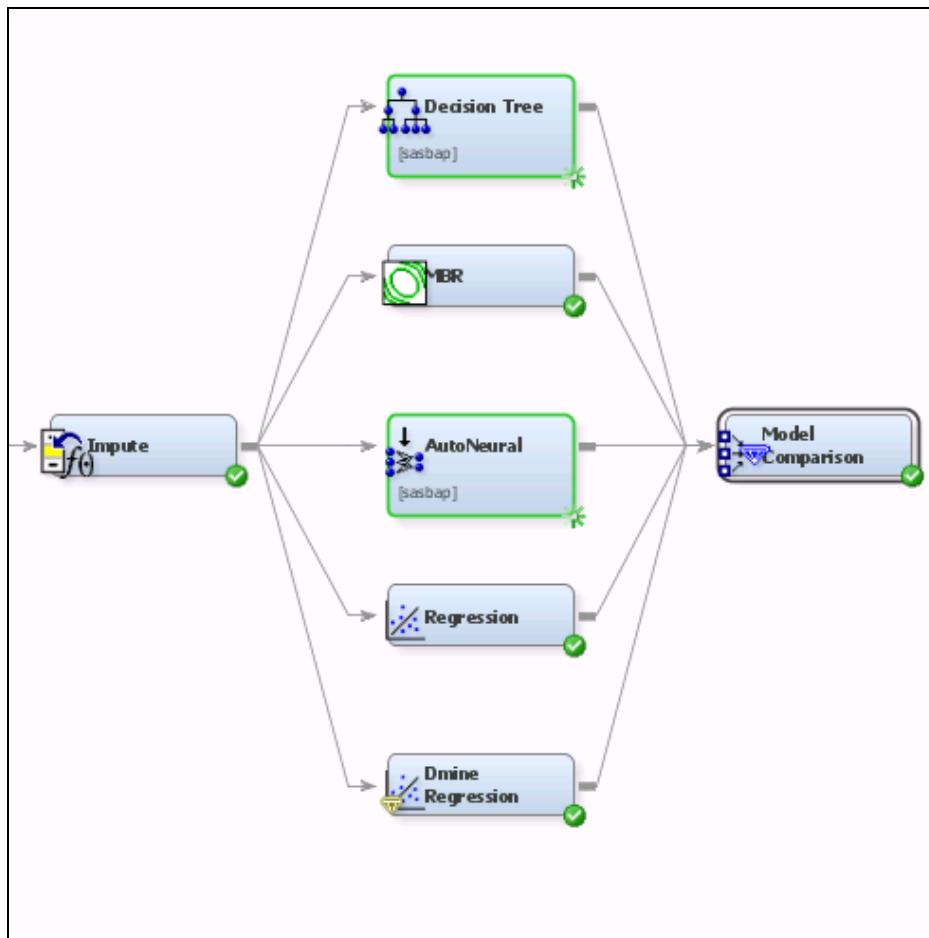


- 3) Click **OK**.

- 4) Right-click the **Model Comparison** node and click **Run**.



- 5) Click **Yes** to confirm. Observe the process in SAS Enterprise Miner.



- 6) Observe the process in Platform RTM.

| Batch Job Filters |                |         |         |                              |                                             |           |              |
|-------------------|----------------|---------|---------|------------------------------|---------------------------------------------|-----------|--------------|
| Cluster:          | All            | User:   | All     | UGroup:                      | All                                         | Status:   | RUNNING      |
| Queue:            | All            | Host:   | All     | HGroup:                      | All                                         | Records:  | 30           |
| JobID:            |                | Search: |         |                              | <input checked="" type="checkbox"/> Dynamic | <b>go</b> | <b>clear</b> |
| << Previous       |                |         |         | Showing Rows 1 to 2 of 2 [1] |                                             |           |              |
| Job               | Job ID         | Name    | Status  | State Changes                | User ID                                     | CPU Usage | CPU Effic    |
| 2123              | SASGrid:316320 |         | RUNNING | 2                            | \chris                                      | 4s        | 14.81%       |
| 2124              | SASGrid:316320 |         | RUNNING | 2                            | \chris                                      | 1s        | 5.88%        |

- 7) Click **OK** when the program finishes processing.  
 8) Close SAS Enterprise Miner.  
 a) Click **File** ⇒ **Exit**.  
 b) Click **Yes** to confirm that you want to close SAS Miner.

**End of Solutions**

## Solutions to Student Activities

### 5.01 Poll – Correct Answer

You do not need a SAS Metadata Server in a SAS Grid Manager environment.

- True
- False

27

False, you must have a SAS Metadata Server in each SAS Grid Manager environment.

### 5.02 Poll – Correct Answer

You can run jobs using the workspace server or the grid server from SAS Data Integration Studio.

- True
- False

35

### 5.03 Multiple Choice Poll – Correct Answer

Which of the following statements is false?

- a. You can set Enterprise Guide projects to use grid sessions.
- b. You can specify that sections of a task use a grid session.**
- c. You can start a grid session with Enterprise Guide when it starts the workspace server.
- d. You can specify that an individual task should use a grid session.

52

### 5.04 Poll – Correct Answer

The SAS Add-In for Microsoft Office can run tasks in parallel on the grid.

- True
- False**

63

# Appendix A Additional Information

|                                        |             |
|----------------------------------------|-------------|
| <b>A.1 SAS Grid Resources.....</b>     | <b>A-3</b>  |
| <b>A.2 Using SAS/CONNECT.....</b>      | <b>A-12</b> |
| Demonstration: Using SAS/CONNECT ..... | A-17        |



# A.1 SAS Grid Resources

## Objectives

- Identify areas of support that SAS offers.
- List additional resources for platform administration.
- Describe resources specific to SAS Grid Manager.

3

## Education

Comprehensive training to deliver greater value to your organization.

- More than 200 course offerings
- World-class instructors
- Multiple delivery methods: instructor-led and self-paced
- Training centers around the world

[support.sas.com/training](http://support.sas.com/training)

4



## SAS Books

Convenient. Practical. Enlightening.

Valuable insight with solid results.



Available in a variety of formats to best meet your needs:

- hard-copy books
- e-books
- PDF

[www.sas.com/store/books](http://www.sas.com/store/books)

5

## SAS Global Certification Program

SAS offers several globally recognized certifications.

- Computer-based certification exams – typically 60-70 questions and 2-3 hours in length
- Preparation materials and available practice exams
- Worldwide directory of SAS Certified Professionals



[support.sas.com/certify](http://support.sas.com/certify)

6

## Support

SAS provides a variety of self-help and assisted-help resources.

- SAS Knowledge Base
- Downloads and hot fixes
- License assistance
- SAS discussion forums
- SAS Technical Support



[support.sas.com/techsup](http://support.sas.com/techsup)

7

## User Groups

SAS supports many local, regional, international, and special-interest SAS user groups.

- SAS Global Forum



- Online SAS community: [www.sascommunity.org/wiki/](http://www.sascommunity.org/wiki/)

[support.sas.com/usergroups](http://support.sas.com/usergroups)

8

## SAS Grid Manager: Classroom Training

The SAS Training page on the web is the best place to find the current training offerings and schedules.

The screenshot shows the SAS Training website interface. At the top, there's a navigation bar with links for support.sas.com, Knowledge Base, Support, TRAINING & BOOKS (which is highlighted in orange), Happenings, Store, and Support Communities. Below the navigation is a search bar with dropdowns for 'Find a course by' and 'Search Courses'. To the right of the search bar is a 'My Training' link. The main content area has a title 'TRAINING & BOOKS / TRAINING' and a sub-section 'TRAINING & BOOKS'. On the left, there's a sidebar with categories like Books, Training, Certification, SAS Global Academic Program, and SAS OnDemand. The main content area features a 'SAS Training' section with a small image of a potted plant and the text 'Free tutorials and e-Learning to get you started.' followed by a 'Learn More' link. Below this is a 'Starting Points' section with several options like 'Get started with SAS', 'Get SAS certified.', etc., each with a corresponding checkbox.

9

[support.sas.com/training](http://support.sas.com/training)

## Next Steps – Possible Courses

To learn more about this: Enroll in the following:

Scheduling



**SAS® Enterprise Scheduling  
with Platform Suite for SAS®**

Grid Administration



**SAS® Grid Manager  
Administration**

10

## SAS Grid Manager Documentation

The SAS Grid Manager documentation can be found on this website:

[support.sas.com/documentation/onlinedoc/gridmgr/index.html](http://support.sas.com/documentation/onlinedoc/gridmgr/index.html)

The screenshot shows the SAS Knowledge Base / Product Documentation page for SAS® Grid Manager. The page features the SAS logo and navigation links for support, training, and store. The main content area displays documentation for SAS Grid Manager 9.4, including links to PDF and HTML versions of Grid Computing and Scheduling guides. A sidebar lists various documentation categories like Products & Solutions, System Requirements, and Documentation.

11

Additional resources on a variety of topics are available:

- PROC SCAPROC
  - [www2.sas.com/proceedings/forum2008/006-2008.pdf](http://www2.sas.com/proceedings/forum2008/006-2008.pdf)
  - [www2.sas.com/proceedings/forum2008/006-2008.pdf](http://www2.sas.com/proceedings/forum2008/006-2008.pdf)
- Importing Programs into SAS Data Integration Studio
  - [support.sas.com/resources/papers/proceedings09/098-2009.pdf](http://support.sas.com/resources/papers/proceedings09/098-2009.pdf)
  - [support.sas.com/rnd/scalability/grid/SGF09\\_DIGrid.pps](http://support.sas.com/rnd/scalability/grid/SGF09_DIGrid.pps)
  - [support.sas.com/rnd/scalability/grid/SGF09\\_SuperDemo\(SCA\).pps](http://support.sas.com/rnd/scalability/grid/SGF09_SuperDemo(SCA).pps)
- SAS Enterprise Guide
  - [support.sas.com/rnd/scalability/grid/SASEGandGrid.zip](http://support.sas.com/rnd/scalability/grid/SASEGandGrid.zip)
  - [support.sas.com/documentation/cdl/en/gridref/62853/HTML/default/n17u3mlv5nrr2n105ipo4vid1ma.htm](http://support.sas.com/documentation/cdl/en/gridref/62853/HTML/default/n17u3mlv5nrr2n105ipo4vid1ma.htm)
- External Workspace Server Load Balancing
  - [support.sas.com/documentation/cdl/en/gridref/62853/HTML/default/n1q6n62da5w2j1n0zl407vg9pg6a.htm](http://support.sas.com/documentation/cdl/en/gridref/62853/HTML/default/n1q6n62da5w2j1n0zl407vg9pg6a.htm)
- %DISTRIBUTE
  - [support.sas.com/rnd/scalability/grid/download.html](http://support.sas.com/rnd/scalability/grid/download.html)

## SAS Papers

SAS Scalability and Performance Community

- [support.sas.com/rnd/scalability/grid](http://support.sas.com/rnd/scalability/grid)

SAS.com

- [sas.com/technologies/architecture/grid/index.html](http://sas.com/technologies/architecture/grid/index.html)

12

## SAS Platform: Classroom Training

The platform for SAS Business Analytics Training page on the web is the best place to find the current training offerings and schedules related to general platform administration.

The screenshot shows the SAS Training Console interface. The top navigation bar includes links for support.sas.com, Knowledge Base, Support, Training & Books (which is highlighted in orange), Happenings, Store, and Support Communities. Below the navigation is a search bar. The main content area is titled "TRAINING & BOOKS / TRAINING". On the left, there's a sidebar with a "Training Console" section containing links for Books, Training (with sub-links for Find a Course, e-Learning, Webinars, Training Formats, Discounts, Free Tutorials, Certification, SAS Global Academic Program, and SAS OnDemand), and SAS Global Academic Program. The main content area is titled "Administration" and describes the SAS Platform. It features a large blue callout box for "Platform Administrator Certification" which includes "Getting Started with the Platform for SAS Business Analytics", "SAS Platform Administration: Fast Track", and "SAS Platform Administration: System Administration". To the right of this, there's another box for "Installing and Configuring the SAS Intelligence Platform" and a link for "NEW! SAS Platform Administration: Middle Tier Administration".

13

[support.sas.com/training/us/patterns/admin.html](http://support.sas.com/training/us/patterns/admin.html)

## SAS Platform: e-Learning

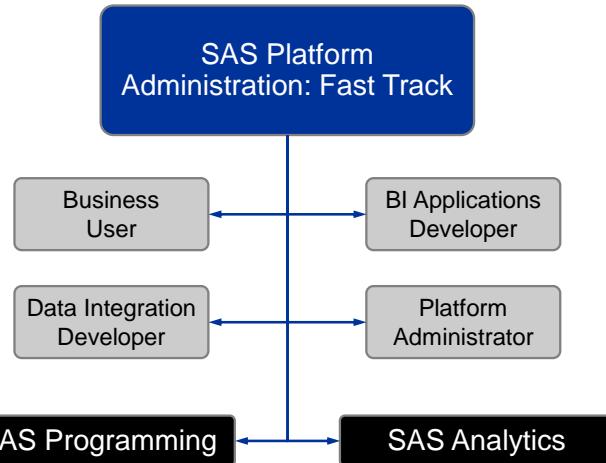
The SAS Self-Paced e-Learning page on the web is the best place to find the current offerings in that format.

The screenshot shows the SAS website's training section. The top navigation bar includes links for support.sas.com, Knowledge Base, Support, Training & Books (which is highlighted in orange), Happenings, Store, and Support Communities. Below the navigation is a breadcrumb trail: TRAINING & BOOKS / TRAINING. The main content area is titled "e-Learning" and has tabs for "Courses" (selected) and "Lectures". It features a sub-section titled "Train when and where you want" with bullet points about eliminating travel costs and enjoying award-winning e-Courses, Recorded Courses, and e-Lectures. Another sub-section titled "Free e-Learning to get you started" lists "SAS Programming 1" and "Statistics 1" with "Start Now" buttons. To the right is a graphic of a lightbulb with a plant growing out of it, labeled "Grow with us". On the left sidebar, under "TRAINING & BOOKS", there are sections for Books, Training (with sub-links for Find a Course, e-Learning, Locations, Training Formats, Discounts, Free Tutorials), Certification, SAS Global Academic Program, and SAS OnDemand.

14

[support.sas.com/edu/elearning](http://support.sas.com/edu/elearning)

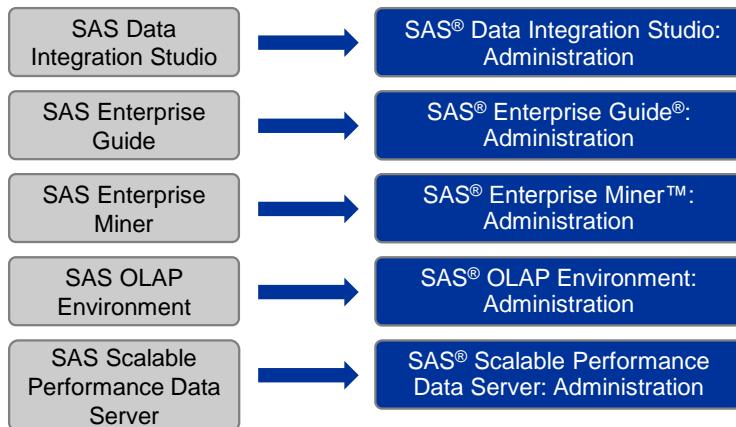
## Next Steps – Job Role Training



15

## Next Steps – Possible Courses

To learn more about this: Enroll in the following:



16

## SAS Intelligence Platform Documentation

The SAS platform administration documentation can be found on this website:

[support.sas.com/documentation/onlinedoc/intellplatform/index.html](http://support.sas.com/documentation/onlinedoc/intellplatform/index.html)

**SAS® Intelligence Platform**

9.4    9.3    9.2    9.1.3

Search SAS 9.4 Intelligence Platform documentation:  
Enter search term

Use the SAS Intelligence Platform documentation to install, configure, and administer a number of offerings such as [SAS Office Analytics](#), [SAS Enterprise BI Server](#), [SAS Data Management](#), and [SAS Data Integration Server](#).

For solutions that are built on the SAS Intelligence Platform (including [SAS Visual Analytics](#)), use this documentation to install, configure, and administer the platform-level components. For additional, solution-level administrative documentation, see the documentation page for your solution.

- What's New in SAS 9.4 Intelligence Platform  
[PDF \(2MB\)](#) | [HTML](#)
- SAS 9.4 Intelligence Platform: Overview  
[PDF \(1.88MB\)](#) | [HTML](#)

17

## SAS Papers

Different types of technical papers can provide the following:

- information for different industries and solutions
- technical details to achieve specific goals and harness robust features of the software

18

## SAS Papers

Different types of papers can be found on the following sites:

- SAS Technical Papers  
[support.sas.com/resources/papers/](http://support.sas.com/resources/papers/)
- SAS Presents  
[support.sas.com/rnd/papers/index.html](http://support.sas.com/rnd/papers/index.html)
- White Papers  
[www.sas.com/whitepapers/index.html](http://www.sas.com/whitepapers/index.html)
- Conference Proceedings  
[support.sas.com/events/sasglobalforum/previous/online.html](http://support.sas.com/events/sasglobalforum/previous/online.html)

19

## A.2 Using SAS/CONNECT

---

### Objectives

- Review SAS/CONNECT statements.
- Use SAS/CONNECT in a grid-enabled program.

21

### SAS/CONNECT

Behind the scenes, SAS/CONNECT is used to submit SAS programs to remote computers. An existing SAS program might need to be updated to include SAS/CONNECT statements in order to use a grid.

22

SAS grid computing provides for distributed computing, where SAS programs submitted by many concurrent users are spread across multiple computer servers in the SAS grid. Users have been able to submit their programs to remote servers using SAS/CONNECT software, but the programs had to be directed to a specifically named server. SAS 8 delivered the capability to run sections of a SAS program in parallel, but the name of the server(s) to run each section of code on were required to be coded into the program.

In SAS grid computing you use the same SAS/CONNECT statement, a signon statement, to request a remote SAS session. The difference in SAS grid computing is that before the signon statement you run the grid enable statement (`grdsvc_enable()`) to inform SAS that the signon statement will use a grid session instead of a SAS/CONNECT session.

Actually, a SAS/CONNECT server session and a grid session are both instances of a SAS/CONNECT server process. The only difference is that a grid session is started by SAS Grid Manager and a SAS/CONNECT server session is started by the SAS/CONNECT spawner. Therefore, after the server session is started, you can use all of the SAS/CONNECT statements and options, excluding the parameters to a signon statement for the hostname of the server where the SAS/CONNECT spawner is running.

## SAS/CONNECT

SAS/CONNECT provides

- cooperative processing for SAS applications
- access to files, hardware, and SAS software on various remote hosts
- parallel processing for independent steps
- piping output from one step to a subsequent dependent step.

## SAS/CONNECT Basics

SAS/CONNECT provides three ways to distribute processing:

- Compute Services
- Data Transfer Services
- Remote Library Services

24

More information about SAS/CONNECT is available in the Distributed and Parallel Processing with the SAS® System course. Information is also available in a series of e-lectures:

- *Accessing a Remote SAS® Data Library*
- *Connecting to a Remote Host, Distributed Processing*
- *Transferring Data and Macro Variables*

When you modify a program to take advantage of a SAS grid, compute services are the focus. Here is an example of a program that uses Data Transfer Services:

```
proc upload data=orion.customers
            out=customers;
    where customer_country="US";
run;
```

Remote Library Services uses the LIBNAME statement to point to a remote library:

```
libname orion '/u/orion/orgold' server=orionunx access=readonly;
```

## SAS/CONNECT Compute Services

Compute Services enable you to move any or all segments of an **application** to other processors to take advantage of hardware, software, and data resources. The additional processor can be on the same machine or on other machines.

25

## SAS/CONNECT Compute Services

To use SAS/CONNECT Compute Services, do the following:

1. Sign on to a remote session.
2. Specify that the task is to run in a separate SAS session with the RSUBMIT statement.
3. Specify the name of the task.
4. Identify that the task is to process asynchronously with the WAIT=NO option.
5. Signal the end of the task with the ENDRSUBMIT statement.
6. Sign off from the remote session.

26

## SAS/CONNECT Compute Services

Examples of Statements:

```
options autosignon;

rsubmit orgold;
  libname orion '/u/orion/orgold';
  proc print data=orion.customer;
    run;
endrsubmit;

signoff orgold;
```



## Using SAS/CONNECT

---

g2d1.sas

```
/* Demo g2d1.sas */
options sascmd='sas -nosyntaxcheck' autosignon;
rsubmit us wait=no sysrputsync=yes;
libname orion 'D:\Grid Demo Docs\SAS Code on Grid\SAS_Code_Extras';

data us_cust;
  set orion.customer;
  where country='US';
run;
/* Retrieve the path location of the Work library */
%sysrput pathus=%sysfunc(pathname(work));
endrsubmit;

rsubmit ca wait=no sysrputsync=yes;
libname orion 'D:\Grid Demo Docs\SAS Code on Grid\SAS_Code_Extras';
options nofmterr;
data ca_cust;
  set orion.customer;
  where country='CA';
run;
/* Retrieve the path location of the Work library */
%sysrput pathca=%sysfunc(pathname(work));
endrsubmit;

waitfor _all_ us ca;
libname uswrk "&pathus";
libname cawrk "&pathca";
options nofmterr;

data concat;
  set uswrk.us_cust cawrk.ca_cust;
run;

proc print data=concat;
run;

signoff _all_;
```

**End of Demonstration**

