

# **How to Interface Your Enterprise Scheduler with SAS® Grid Manager**

**January, 2013**

## Table of Contents

<b>Introduction.....</b>	<b>3</b>
Scope.....	3
Terms.....	3
<b>Integration Details.....</b>	<b>4</b>
<b>Appendix: Contents of the Helper Scripts .....</b>	<b>6</b>
SAS 9.4 and later.....	6
SAS 9.3M2.....	6
SAS 9.2M2 through 9.3M1 .....	7
SAS 9.3M0 and earlier.....	8

# Introduction

---

It is commonly the case that SAS customers use an enterprise scheduler that has been chosen by their IT organizations as their standard scheduler. Common examples would include BMC Control-M and CA 7. In addition they have considerable investment in their enterprise scheduler with the number of jobs that already use the scheduler, the existing knowledge of the scheduler by their users as well as integration with incident management and accounting systems. It is also the case that when deploying SAS in a SAS Grid environment, there is the need to be able to submit all SAS workload to the Grid. However, it is typically cost prohibitive and impractical for such a customer to learn and convert to the scheduling capabilities that are part of the Platform Suite for SAS that is shipped with SAS Grid Manager. This paper details the few steps necessary to interface an existing enterprise scheduler with SAS Grid Manager so that scheduled jobs are transparently submitted to the grid with no change to the way the existing enterprise scheduler is used.

---

## Scope

This document details the steps required to interface an enterprise scheduler with SAS® Grid Manager via a UNIX or Windows command line interface. This information applies to any enterprise scheduler and does not require any special features or functionality from the scheduler.

---

## Terms

- **SAS Grid Manager Client Utility** – a command-line utility (**SASGSUB**) that enables users to submit SAS programs to a grid for batch processing.
- **SAS Job** – a .sas file containing a SAS program.
- **Helper script** – a UNIX or Windows shell file which expands on the functionality of an existing program.

# Integration Details

The goal of this integration is to maintain the way that a customer uses their existing scheduler by configuring SAS such that the scheduled jobs are transparently submitted to SAS Grid Manager. This requires the following:

- Modify the SAS invocation that is used by the enterprise scheduler to do a grid submission. In most cases this will be done by calling SASGSUB to submit the job to the grid. In order to limit the redirection of the SAS invocation to scheduled jobs only, change the scheduler machine's system path to redirect any SAS call to a helper script which will in turn call SASGSUB instead of the SAS executable.
- Wait for the job to finish and return the job RC back to the enterprise scheduler.

The following steps provide the integration details:

1. Create a directory on the machine where your scheduler invokes SAS. The helper script will be placed in this directory. Use the appropriate helper script for the version of SAS being used as indicated in the table below. The scripts call SASGSUB with the exception of the 9.2M1 and earlier scripts. Make sure to configure sasgsub.cfg to reflect the appropriate options for your enterprise scheduler. The scripts can be downloaded from:

<http://support.sas.com/rnd/scalability/grid/download.html>

SAS Version	UNIX Script	Windows Script
9.4+	No script needed.	No script needed.
9.3M2 and later	gsubrc93m2.sh	gsubrc93m2.cmd
9.2M2 to 9.3M1	gsubrc93.sh	gsuvrc93.cmd
9.2M1 and earlier	runsas.sh	runsas.cmd

2. Modify the helper script to set the correct paths. For UNIX, the location of the LSF configuration script as well as the location of the SAS configuration directory must be specified. For Windows, the location of the SAS configuration directory must be specified.
3. For UNIX, create a file named "sas". For Windows, create a file named "sas.cmd." Modify the contents of this file as indicated below for the appropriate operating system and release of SAS. This file will replace the SAS executable for your scheduler. All helper scripts expect the first command line parameter to be the SAS file to be run. If you specify a relative (not fully qualified) path, it must be relative to the directory which contains the helper script.

**NOTE:** If SASGSUB is used, it is assumed that the credentials needed to access the SAS metadata server are specified in the sasgsub.cfg file. METAPASS should be changed from \_PROMPT\_ to the encrypted value of the password corresponding to the METAUSER value.

## UNIX (9.4+)

Add the following line to the "sas" file and change the path to the location of SASGSUB on your machine.

```
. /path/to/sasgsub -GRIDWAITRESULTS -GRIDSUBMITPGM $1 <SASGSUB OPTIONS>
```

### UNIX (9.2M2 through 9.3M2)

Add the following line to the “sas” file:

```
./<HELPERSCRIPTNAME> $1 <SASGSUB OPTIONS>
```

where <HELPERSCRIPTNAME> is either gsubrc93 or gsubrc93m2

### UNIX (9.2M1 and earlier)

Add the following line to the “sas” file:

```
./runsas.sh $1 <ADDITIONAL SAS OPTIONS>
```

The 9.2M1 and earlier wrapper script requires some modification to operate correctly. It does not call SASGSUB so the sasgsub.cfg options will not affect this script. This wrapper script uses SAS/CONNECT to submit the job to the grid. Therefore, SAS/CONNECT would have to be licensed on the scheduler machine.

### Windows (9.4+)

Add the following line to the “sas.cmd” file and change the path to the location of SASGSUB on your machine. SASGSUB is usually located in the SAS configuration directory under “Applications\SASGridManagerClientUtility\9.4.”

```
CALL C:\Path\To\SASGSUB -GRIDWAITRESULTS -GRIDSUBMITPGM %1 <SASGSUB OPTIONS>
```

### Windows (9.2M2 through 9.3M2)

Add the following line to the “sas.cmd” file:

```
CALL <HELPERSCRIPTNAME>.cmd %1 <SASGSUB OPTIONS>
```

where <HELPERSCRIPTNAME> is either gsubrc93 or gsubrc93m2

### Windows (9.2M1 and earlier)

Add the following line to the “sas.cmd” file:

```
CALL runsas.cmd %1 <ADDITIONAL SAS OPTIONS>
```

The 9.2M1 and earlier wrapper script requires some modification to operate correctly. It does not call SASGSUB so the sasgsub.cfg options will not affect this script. This wrapper script uses SAS/CONNECT to submit the job to the grid. Therefore, SAS/CONNECT would have to be licensed on the scheduler machine.

4. Add the new ‘alternate’ directory into the system path or your scheduler’s startup path **before** the existing SAS path entries. All calls to SAS will now go to the newly created “sas” or “sas.cmd” script and be submitted to the grid via the helper script.

## Appendix: Contents of the Helper Scripts

---

This appendix provides additional information specific to the helper scripts that can be downloaded from the Scalability site.

---

### SAS 9.4 and later

The SAS® Grid Manager Client Utility for SAS 9.4 has an option to wait for a job to complete and to return the return code of the job. No helper script is required to use the SAS Grid Manager Client Utility with SAS 9.4, simply use the `-GRIDWAITRESULTS` option as indicated below:

```
. /path/to/sasgsub -GRIDWAITRESULTS -GRIDSUBMITPGM <SAS FILE>.sas  
<SASGSUB OPTIONS>
```

The SAS Grid Manager Client Utility will wait for the job to finish and print the job's return code. Output files will be found in a new subdirectory of the execution environment

---

### SAS 9.3M2

The SAS Grid Manager Client Utility for SAS 9.3M2 provides an option to wait for a job to complete however it does not print a job's return code. Therefore, a helper script called **gsubrc93m2** is used to submit a program through the SAS Grid Manager Client Utility, wait for completion, and return the return code of the job.

#### Unix

The script, **gsubrc93m2.sh**, can be placed in any directory on your UNIX machine; it does not have to be in the same directory as the SAS Grid Manager Client Utility. You must edit the indicated variables at the top of the script. You will need to set the location of the LSF configuration script and the SAS configuration directory. Make sure that the necessary permissions are set so that both you and your scheduler may execute the file with a standard BASH shell – this may require a change in both user and folder permissions.

#### Windows

The windows helper script, **gsubrc93m2.cmd**, can be placed in any directory on your Windows machine; it does not have to be in the same directory as the SAS Grid Manager Client Utility. You must edit the path variable where indicated in the script to set the correct location of the SAS configuration directory.

Please note that **gsubrc93m2.cmd** requires that CMD.EXE's command extensions be turned on. This is on by default in Windows XP and subsequent releases of the Windows OS.

#### Usage

The syntax of **gsubrc93m2** is as follows:

UNIX:

```
gsubrc93m2.sh $1 <SASGSUB options>
```

Windows:

```
gsubrc93m2 %1 <SASGSUB options>
```

where:

\$1 (or %1) is required. It represents a relative or fully qualified path to the SAS job you want to submit to the grid.

<SASGSUB options> are optional. They represent any additional options to pass through to the SAS Grid Manager Client Utility. For more information on those options, please see [Submitting Batch SAS Jobs to the Grid](#).

On Unix/Linux systems the exit code (\$?) of `gsubrc93m2.sh` will be the return code of the submitted job. On Windows systems the %ERRORLEVEL% environment variable will be set to the return code of the job. Both UNIX and Windows scripts will copy various output files to their execution directory before exiting. Please note that \*.lst files are more commonly created on UNIX systems than windows systems and are not guaranteed to exist when these helper scripts exit.

---

## SAS 9.2M2 through 9.3M1

The SAS Grid Manager Client Utility for SAS 9.2M2 through 9.3M1 does not provide the ability to wait for a job to complete and does not print a job's return code. Therefore, a helper script called `gsubrc93` is used to submit a program through the SAS Grid Manager Client Utility, loop until the job has completed, and return the return code of the job.

### Unix

The script `gsubrc93.sh` can be placed in any location on your UNIX machine; it does not have to be in the same directory as the SAS Grid Manager Client Utility. You must edit the indicated variables at the top of the script. You will need to set the location of the LSF configuration script and the SAS configuration directory. Make sure that the necessary permissions are set so that both you and your scheduler may execute the file with a standard BASH shell – this may require a change in both user and folder permissions.

### Windows

The windows helper script, `gsubrc93.cmd`, can be placed in any directory on your Windows machine; it does not have to be in the same directory as the SAS Grid Manager Client Utility. You must edit the path variable where indicated in the script to set the correct location of the SAS configuration directory.

Please note that `gsubrc93.cmd` uses advanced features of the Windows command processor such as extensions and delayed expansion. Though enabled by default, if these settings are disabled on your system the helper script will not execute correctly.

### Usage

The syntax of `gsubrc93` is as follows:

UNIX:

```
gsubrc93.sh $1 <SASGSUB options>
```

Windows:

```
gsubrc93 %1 <SASGSUB options>
```

where:

\$1 (or %1) is required. It represents a relative or fully qualified path to the SAS job you want to submit to the grid.

<SASGSUB options> are optional. They represent any additional options to pass through to the SAS Grid Manager Client Utility. For more information on those options, please see [Submitting Batch SAS Jobs to the Grid](#).

On Unix/Linux systems the exit code (\$?) of gsubrc93.sh will be the return code of the submitted job. On Windows systems the %ERRORLEVEL% environment variable will be set to the return code of the job. Both UNIX and Windows scripts will copy various output files to their execution directory before exiting. Please note that \*.lst files are more commonly created on UNIX systems than windows systems and are not guaranteed to exist when these helper scripts exit.

---

## SAS 9.2M1 and earlier

### *Using a general purpose wrapper script*

For SAS 9.2M2 and later Grid deployments, SASGSUB should be used with the helper scripts described earlier in this paper. Prior to the existence of SASGSUB, SAS created a general purpose wrapper script so that customers who are running SAS 9.2M1 and earlier could interface their enterprise scheduler with SAS Grid Manager. This wrapper script uses SAS/CONNECT to submit the job to the grid. Therefore, SAS/CONNECT would have to be licensed on the scheduler machine. You can download the wrapper script here:

Before using the wrapper script, the following must be specified where indicated in the script:

- SAS\_HOME should be set to the correct SAS path on your computer (For UNIX, usually "/usr/local/SAS/SAS\_9.x")
- SAS\_METASERVER, SAS\_METAPORT, SAS\_METAREPOS, SAS\_METAUSER, and SAS\_METAPASS should reflect the correct connection information for the metadata server at your deployment. The default SAS\_METAPORT is 8561 and the default SAS\_METAREPOS is 'Foundation' (quotes included).
- SAS\_GRIDSERVER should be set to the name of the application server associated with the logical grid server.

Make sure that you and your scheduler have the appropriate user and folder permissions to execute the wrapper script. Once you have configured the wrapper script, you can call it with the following arguments directly from your scheduler:

### **Usage**

The syntax of runsas is as follows:

UNIX:

```
runsas.sh $1 <additional SAS options>
```

Windows:

```
runsas %1 <additional SAS options>
```

where:

\$1 (or %1) is required. It represents a relative or fully qualified path to the SAS job you want to submit to the grid.

The <additional SAS options> are optional. They represent any additional SAS options to pass to the SAS session running on the grid.