



THE
POWER
TO KNOW.

SAS[®] Enterprise Miner[™] 6.2 Administrator's Guide



The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2010. *SAS® Enterprise Miner™ 6.2: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Enterprise Miner™ 6.2: Administrator's Guide

Copyright © 2010, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hard-copy book: 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.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

U.S. Government Restricted Rights Notice: Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227-19, Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st electronic book, August 2010

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at support.sas.com/publishing or call 1-800-727-3228.

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 registered trademarks or trademarks of their respective companies.

SAS Enterprise Miner 6.2: Administrator's Guide

- [SAS Enterprise Miner 6.2 Planning and Preparation](#)
 - [Migrating from SAS Enterprise Miner 6.1 to SAS Enterprise Miner 6.2](#)
 - [SAS Enterprise Miner 6.2 Client/Server Sizing](#)
 - [SAS Enterprise Miner 6.2 Single User and Multi-User Deployments](#)
 - [SAS Enterprise Miner 6.2 Plug-in for SAS Management Console](#)
 - [Adding a New Enterprise Miner User Identity in the SAS 9.2 Management Console](#)
 - [Creating Different Workspace Servers in SAS Management Console for Enterprise Miner Project Segregation](#)
 - [SAS Enterprise Miner 6.2 Client Startup Properties](#)
 - [SAS Enterprise Miner 6.2 Projects](#)
 - [SAS Enterprise Miner 6.2 Models](#)
 - [Allocating Libraries for SAS Enterprise Miner 6.2](#)
 - [SAS Enterprise Miner 6.2 PMML Support](#)
-

[SAS Enterprise Miner 6.2 Planning and Preparation](#)

- [Network Performance](#)
 - [Mapped Network Drives](#)
 - [Multitasking](#)
 - [Threading](#)
 - [SAS Servers](#)
 - [Java Requirements](#)
 - [SAS Analytics Platform Midtier](#)
 - [SAS Enterprise Miner Client](#)
 - [Batch Processing](#)
-

[Network Performance](#)

All SAS computations are performed on the server and are unaffected by the client/server network performance. All project information such as functional settings and intermediate data sets are stored on the SAS server. GUI operations (such as editing process flow diagrams, property sheets, and variables tables) depend on transferring data from the SAS server to the client GUI and are affected by both the server's availability and client/server network performance. SAS Enterprise Miner 6.2 is designed to tolerate network disconnections, and will clean up resources that include SAS sessions. A reliable 1-GB or greater network bandwidth is recommended for baseline client/server performance.

[Mapped Network Drives](#)

SAS Workspace and Stored Process servers are not accessible through Windows mapped network drives. You cannot use a Windows mapped drive to create or to access Enterprise Miner projects on a network or access remote libraries if a Windows mapped drive is specified in the path. If an Enterprise Miner Windows client needs access to SAS resources or project libraries on a networked location, the path to the resource should be specified using the network Universal Naming Convention (for example, `\\mypc\myshare\libraries`).

[Multitasking](#)

The SAS Enterprise Miner 6.2 client starts a SAS session for interacting with the server and submitting SAS code. Each node in a process flow diagram also runs in its own SAS session. Each user can therefore start many SAS sessions on the data mining server.

Note: It is possible for you to start so many SAS sessions on a single CPU desktop system that overall system performance is seriously degraded.

[Threading](#)

Some tasks in SAS Enterprise Miner 6.2 (such as data sorting, variable selection, and regression modeling) have been built to distribute their work over multiple CPUs on the same system. Refer to the SAS

SAS Servers

The SAS Enterprise Miner 6.2 server is responsible for project storage and computations. To use the SAS Enterprise Miner 6.2 server, you must have installed both a SAS Object Spawner and a SAS Metadata server. Most installation plans will include creation of these services.

Java Requirements

The SAS Enterprise Miner 6.2 client requires Java release 1.5.12 in Windows operating environments and release 1.5.12 in UNIX operating environments. Installed versions of SAS 9.2 include Java 1.5.12. No further version of Java is required.

The SAS Enterprise Miner Client can be delivered to users through Java WebStart. In this case, you will need to download and install the correct version of Java to your local system.

SAS Analytics Platform Midtier

You can install the SAS Analytics Platform as a midtier service that will serve data mining functionality to multiple users. In this configuration, you can connect and disconnect from long running processes. This is the recommended configuration when working with remote servers.

SAS Enterprise Miner Client

You should be aware of the following considerations when using the SAS Enterprise Miner 6.2 client.

- **Memory** — The maximum Enterprise Miner Java client memory is set to 512 MB. Client systems should have 512 MB of memory for best performance.
 - **Property Sheets and Variables Tables** — The data for property sheets and variables table editors is retrieved from the server. If the server is running a resource-intensive diagram, the property sheet or the variables table might be slow to appear.
 - **Program Editor** — User-entered code that is submitted through the Program Editor runs in the same SAS session as other GUI operations. Submitting code through the Program Editor blocks other GUI operations, such as concurrently editing a process flow diagram or a property sheet, and interacts in the same way as the traditional SAS Editor session.
 - **Graphics and Table Views** — Data for interactive graphics and table views is transferred on demand from the server to the client. The maximum number of rows of data that can be transferred from the server to the client depends on the data. For interactive graphics, you can select a sampling method and size. SAS variable formats are not currently supported in interactive graphics and table views.
-

Batch Processing

SAS Enterprise Miner 6.2 supports SAS language-based batch processing by using macros. You can use all these macros together in one or in many SAS jobs that conduct complete data mining model building without using the GUI in SAS Enterprise Miner 6.2, or you can use the macros with the GUI in a complementary cycle.

- **%EMDS** — supports the creation of SAS Enterprise Miner 6.2 data source definitions and management of table and column metadata. This capability is useful in automating routine data source registration from batch data preparation jobs. For example, you might have a SAS code job that extracts data from relational tables in a data warehouse and creates a table that is ready for predictive modeling. You can add the %EMDS macro to that job to create the data source definition that is needed for use in the SAS Enterprise Miner 6.2 user interface or in batch computing jobs.
 - **%EMTP** — supports the creation of SAS Enterprise Miner 6.2 target profile definitions that are properties of a data source definition. These target profile definitions contain information about the decision matrix, cost values, and prior probabilities, that are used by Enterprise Miner modeling functions.
 - **%EMSBATCH** — supports the building, running, and reporting of SAS Enterprise Miner 6.2 process flow diagrams. An exact diagram can be run from either the SAS Enterprise Miner 6.2 user interface or from a batch job. The results can be viewed in the SAS Enterprise Miner 6.2 user interface or included in a reporting SAS program. The SAS Enterprise Miner 6.2 user interface includes a function to build batch code. This is useful when you want to automate creation and execution of a data mining analysis.
-

[Migrating from SAS Enterprise Miner 6.1 to SAS Enterprise Miner 6.2](#)

For information on migrating from SAS Enterprise Miner 6.1 to SAS Enterprise Miner 6.2, see <http://support.sas.com/software/maintenance/>.

[SAS Enterprise Miner 6.2 Client/Server Sizing](#)

- [Introduction](#)
 - [What Size Server Do I Need?](#)
 - [How Much Memory?](#)
 - [How Many Processors?](#)
 - [How Much Disk Space?](#)
 - [Computer Resources Needed](#)
 - [Conclusion](#)
 - [Appendix A: Tuning Parameters](#)
 - [Appendix B: File System Suggestions](#)
 - [Appendix C: Nodes that Create Duplicate Copies of Data File](#)
-

[Introduction](#)

This document discusses required resources for data mining using SAS Enterprise Miner 6.2. Understanding the resource requirements helps you to size a Windows or UNIX server for your projects.

This document assumes that the server is dedicated to Enterprise Miner 6.2 only. If other applications will be running on the server, then the information mentioned needs to be added to the needs of the other applications. Please note that Enterprise Miner 6.2 requires SAS 9.2.

[What Size Server Do I Need?](#)

The answer to this question is *it depends* _____ on the following:

- How many Enterprise Miner 6.2 projects will be running at any given time?
- How much data will these Enterprise Miner 6.2 projects be accessing?
- How many variables will be analyzed?
- What is the expected response time of the users?
- Will there be other applications, such as other SAS sessions or RDBMSs running on the server?

The multithreaded procedures in Enterprise Miner 6.2 will use all the CPUs on the server. The multitasking process flow diagrams will use all the CPUs on the system. Enterprise Miner 6.2 users may also start multiple training runs concurrently. If caution is not exercised, users can start too many concurrent processes. If the concurrent process load is too high, the system spends too much time swapping data, which significantly slows down computation times.

The following factors affect Enterprise Miner 6.2 server sizing:

- Large numbers of input variables require more memory.
- Large numbers of categorical variables, especially those with many levels (such as ZIP codes) require extra memory during modeling. Categorical variables are memory intensive because some data mining functions create separate dummy variables for each unique level of the categorical input.
- Some modeling methods for training data can be memory intensive.
- If you have an RDBMS running on the server (especially if it is supporting an OLTP system), then there may be file system resource conflicts between the RDBMS and Enterprise Miner 6.2. The operating system and RDBMS should be configured to operate on large page sizes. Using small page sizes with Enterprise Miner 6.2 bottlenecks the throughput of mining data and adds significant computing time to model training.

The remaining sections provide recommendations for the amount of memory, the number of processors, and the amount of disk space needed to perform data mining with Enterprise Miner 6.2.

[How Much Memory?](#)

Memory for Client and Server Machines

The Enterprise Miner 6.2 product is designed to run in a client/server environment. Here are

some recommendations for memory on Enterprise Miner 6.2 client and server machines.

Client — a minimum of 512 MB of physical RAM dedicated to the Enterprise Miner 6.2 product running on the client machine.

Server — a minimum of 512 MB of physical RAM per processor.

Configuring the MEMSIZE Parameter

After sizing and configuring memory for your server, you must allocate memory for the Enterprise Miner 6.2 application. Use the MEMSIZE parameter in the config.sas file to specify the memory allocation for Enterprise Miner 6.2. The config.sas file is located in the root of your Enterprise Miner 6.2 installation directory.

The MEMSIZE option is the upper limit to the amount of memory the SAS System can use on a per-session basis. Each session uses memory as required up to the specified MEMSIZE limit. The configuration file is located in the root directory of your Enterprise Miner 6.2 installation.

Windows Server — The MEMSIZE parameter is set to 0 by default, which enables each SAS session to take all available memory on the server. The MEMSIZE parameter recommendation is 512 MB.

UNIX Server — The MEMSIZE parameter is set to a given value by default. The value varies according to different UNIX versions of SAS. Depending on the memory requirements for your project, it might be necessary to increase the value specified for the MEMSIZE option. The MEMSIZE parameter recommendation is the amount of RAM equal to 80% of available host memory. Refer to the host documentation for more information on this option.

Never set the MEMSIZE option to a value that is too high. If multiple SAS sessions are running and every SAS session consumes the maximum memory allotted by MEMSIZE, it is possible to exceed the amount of physical RAM on some servers. On Windows servers, the sum of Enterprise Miner 6.2 MEMSIZE allocations for the peak concurrent users should never exceed the physical amount of RAM, as such conditions result in severe I/O bottlenecks.

Refer to Appendix B: Tuning Parameters for more information related to this topic.

Configuring the CATCACHE Parameter

The CATCACHE parameter is a SAS system parameter that controls whether SAS catalogs are kept open in cached memory for repeated use instead of opening and closing the SAS catalogs each time SAS calls them. Enterprise Miner Java clients are not compatible with SAS system CATCACHE settings that are greater than zero. For best Enterprise Miner performance, the CATCACHE parameter should be set to CATCACHE=0 or CATCACHE=MIN in the config.sas file for the SAS session that launches Enterprise Miner.

How Many Processors?

The modeling phase of an Enterprise Miner 6.2 data mining project can require intense memory, CPU, and I/O resources. A good rule of thumb for Windows and UNIX servers is one CPU is capable of supporting between 2 and 5 concurrent Enterprise Miner 6.2 projects.

The underlying procedures of the Association node can be even more disk-space- and memory-intensive than the modeling nodes, especially where thousands of items exist in the transaction data set.

The nature of the data that is typically mined influences the processor needs:

- If a data set has a large number of input variables, more resources are consumed and the server will support fewer concurrent sessions.
- If class variables have many unique values, more resources are consumed and the server will support fewer concurrent sessions.
- If a data set uses a small number (such as 10-15) of input variables that have relatively few unique values, fewer resources are consumed and the server will support more concurrent sessions.

How Much Disk Space?

Disk space requirements should be analyzed for both the client and server machines. In addition to the amount of disk space needed to install Enterprise Miner 6.2 (see Appendix A), you will need adequate disk space to store the following:

- The actual data to be mined. You must allocate space for the source data, whether it is SAS data sets or RDBMS tables.
- The work data that is generated during Enterprise Miner 6.2 processing on the server. A general rule of thumb is for each project to allow for 4-7 times the amount of the data file being analyzed.

- The Enterprise Miner 6.2 Sample, Data Partition, and Memory-Based Reasoning nodes create work files that are as large as the input data files during execution. If your data analyses use these nodes regularly, plan your disk space allocation accordingly.

If you use RAID 1 or RAID 5, your Enterprise Miner 6.2 disk space requirements are over and above any space required for the operating system and the RAID file system.

Performance Tips for Disk I/O

To get the best performance with your disks, you need to balance the I/O subsystem on your server. Some general tips for balancing the I/O subsystem are as follows:

- Change the default location for the SASWORK temporary directory to a file system other than the one that the operating system uses for swap files. The default location for SASWORK uses the same file system for both temporary disk space and for the swap file.
- Place all the heavy I/O activity directories (SASWORK, Enterprise Miner Project Libraries) on separate file systems, using independent I/O paths when possible.

Some modeling nodes create enormous temporary files in the SASWORK directory. Be sure that your SASWORK directory is at least 1.5 – 2 times as large as your data set.

Computer Resources Needed

The processor and memory resources required to perform a data mining analysis depend on the number of observations and variables in the input data, in the complexity of the data model, and in the training algorithm used. For many modeling algorithms, there is a trade-off between time and memory.

In order for an Enterprise Miner 6.2 modeling node to run, there must be enough free memory to support the operating system, the SAS Supervisor, and the Enterprise Miner software while maintaining a free overhead of about 20 to 30 megabytes.

Computer resources can be calculated using a formula:

N	the number of cases (anywhere from 100 to 100,000,000 cases are typical).
V	the number of input variables.
I	the number of input terms or units, including dummy variables, interactions, and polynomials.
W	the number of weights in a neural network.
Q	the number of output units.
D	the average depth of a tree.
R	the number of times the training data is read in logistic regression or neural nets, which depends on the training technique, the termination criteria, the model, and the data. R is typically much larger for neural nets than for logistic regression. In regard to training techniques, R is usually smallest for Newton-Raphson or Levenberg-Marquardt, larger for quasi-Newton, and still larger for conjugate gradients.
S	the number of steps in stepwise regression, or 1 if stepwise regression is not used.

Decision Tree Node — The minimum additional memory required is **8N** bytes. Decision Tree training is considerably faster if the entire data set can be loaded into available RAM. You should have about **[8N*(V + 1)]** bytes of free RAM. If the data will not fit in RAM memory, it is written to a utility file. Memory is also required to hold summary statistics and other tables generated as part of a node's output, but the

space required for summary statistics and tables is much smaller than the amount required to train the decision tree.

Regression Node — The required memory depends on the model type and on the selected training technique. During linear regression, the SSCP matrix dominates memory usage. SSCP memory usage is estimated as $[8 \cdot I^2]$ bytes. During logistic regression, the selected training technique drives memory usage. See the [SAS/OR Software Technical Report: The NLP Procedure](#) to view memory allocation by training technique. Memory requirements range from approximately $[40 \cdot I]$ bytes for the Conjugate Gradient technique to about $[8 \cdot I^2]$ bytes for the Newton-Raphson technique.

Neural Network Node — The memory usage depends on the selected training technique. See the [SAS/OR Software Technical Report: The NLP Procedure](#) to view memory allocation by training technique. Memory requirements range from approximately $[40 \cdot W]$ bytes needed for the Conjugate Gradient technique, to as many as $[4 \cdot W^2]$ bytes needed for the Quasi-Newton and Levenberg-Marquardt techniques.

You can calculate W for a neural network with biases and H hidden units in one layer:

$$W = [H \cdot (I + 1)] + [Q \cdot (H + 1)].$$

For both logistic regression and neural nets, the Conjugate Gradient technique, which requires the least memory, must usually read the training data many more times than the Newton-Raphson and Levenberg-Marquardt techniques.

Assuming that the number of training cases is greater than the number of inputs or weights, the time required for training is roughly proportional to the following:

NI²	for linear regression
SRNI	for logistic regression using conjugate gradients
SRNI²	for logistic regression using quasi-Newton or Newton-Raphson. Note that R is usually considerably less for these techniques than for conjugate gradients
DNI	for tree-based models
RNW	for neural nets using conjugate gradients
RNW²	for neural nets using quasi-Newton or Levenberg-Marquardt. Note that R is usually considerably less for these techniques than for conjugate gradients

Association Node — Memory usage is proportional to $[N^M]$, where N is the number of unique items in the data, and M is the maximum number of items in an association. Both lowering M and/or using higher confidence and/or higher support values can reduce memory and disk usage requirements when performing an associations analysis in Enterprise Miner 6.2.

Conclusion

This information is a guideline to understanding resource requirements for Enterprise Miner 6.2. There are several key sizing factors. One is the size of the typical input data set used for data mining. Data set size is affected by the number of observations, the number of variables, and the number of unique levels. The chosen modeling algorithm and its underlying complexity also help to quantify the amount of resources needed to run an Enterprise Miner 6.2 data mining project.

If you have any additional questions about hardware or software configuration, please contact your SAS Account Executive. Your SAS Account Executive can arrange a conference call with the SAS Technology Center to address your specific questions.

Appendix A: Tuning Parameters

- **Change the default MEMSIZE and SORTSIZE parameters** — Change the MEMSIZE and SORTSIZE parameters on the client and server installations of Enterprise Miner 6.2. Set the MEMSIZE parameter to 80% of total host RAM and the SORTSIZE parameter to 16 MB. The MEMSIZE and SORTSIZE parameter definitions should be submitted in the config.sas file typically located in the root directory of your Enterprise Miner 6.2 installation.
 - **Move the default location of the SASWORK directory** — The default config.sas file locates the SASWORK directory in a directory that exists on all platforms. In its default location, the SASWORK directory is the same directory that the operating system uses to store its swap files. Moving the SASWORK directory to a different disk or file system will result in performance improvements. If possible, avoid using RAID-1 or RAID-5 fault-tolerant file systems, because the RAID algorithms can require additional disk writes.
 - **Run the UNIX SAS jobs with the FULLSTIMER option turned on** — When running your Enterprise Miner 6.2 projects on UNIX, you can add the OPTIONS FULLSTIMER statement to the project parameters. When you add the OPTIONS FULLSTIMER statement, SAS gathers statistics on the resources that it needs and uses to run the application. Examining the FULLSTIMER statistics can help you detect I/O bottlenecks and to quantify the memory needed to run the application. After you tune your resources, turn the FULLSTIMER option back off for even greater resource utilization.
 - **Give Enterprise Miner 6.2 Projects all the memory they need** — If you run a data mining analysis with the FULLSTIMER option specified, you can determine how much allocated memory you are using. Performance improves if you increase the amount of memory Enterprise Miner 6.2 can access. To increase the allocated memory, modify the MEMSIZE option in the config.sas file. The config.sas file is located in the root of your Enterprise Miner 6.2 installation directory.
 - **Monitor CPU for I/O bottlenecks** — Use various UNIX command line tools like vmstat, iostat, ps and sar; as well as GUI based tools like Solstice SYMON and proctool from Sun Microsystems; Performance Manager for Tru64 (Digital UNIX) from Compaq; and Glance from Hewlett-Packard to do performance monitoring on the UNIX server. For Windows servers, you can use a GUI based tool called performance monitor (perfmon) from Intel to gather the same information on Windows servers. If you are seeing lots of disk activities, you might need to reorganize your file systems on the server to get the best I/O performance.
 - **Balance MEMSIZE vs. physical memory** — The optimal memory configuration is to ensure that the summed MEMSIZE values of all concurrent Enterprise Miner 6.2 project users at peak times are less than the physical memory configured on the server.
-

[Appendix B: File System Suggestions](#)

- **Configure sequential I/O for large blocks** — Use the SAS BUFSIZE option to specify a large stripe size such as 64K. However, if your typical data mining task uses indexes that have high cardinality to return small result sets, you should leave the BUFSIZE option in its default setting. In this case, the default BUFSIZE performs random access I/O at the file system level.
 - **Separate input file systems from output file systems when Enterprise Miner 6.2 transforms data with little reduction** — Enterprise Miner 6.2 DATA steps that produce transformations with little reduction of the incoming data, SQL statements that perform similar operations, and APPEND operations that add data to the end of another data file are all disk-intensive tasks. If you can locate Enterprise Miner 6.2 input, output, and work data sets across separate disks and/or disk controllers, you will reduce the potential for I/O bottlenecks.
 - **Set up separate file systems for each user** — Enterprise Miner 6.2 creates lots of intermediate files that are stored in the PROJECT library. If you have lots of Enterprise Miner 6.2 users, you may want to set up several different file systems and then partition the users between these files systems in as balanced a manner as possible. This should reduce the number of FSYNCs issued by the SAS System.
-

[Appendix C: Nodes That Create Duplicate Copies of Data Files](#)

Most nodes in an Enterprise Miner 6.2 project use a common data file and create only small tables. However, three nodes can make a second copy of the input data file each time they are used. These additional copies are stored in the data library on the server. The nodes are as follows:

- Data Partition node
 - Sampling node (but only the amount sampled)
 - Memory Based Reasoning (MBR) node
-

[SAS Enterprise Miner 6.2 Single-User and Multi-User Deployment](#)

There are two ways to deploy SAS Enterprise Miner:

- **Personal workstation:** When Enterprise Miner is installed in the personal workstation configuration, Java files, configuration files, and documentation are stored locally on the client computer. The client computer communicates directly with the SAS Workspace Server and the SAS Metadata Server. The client computer must remain connected to the SAS Workspace Server and the SAS Metadata Server during any model training event. The personal workstation installation is appropriate for single-user deployments.
- **Enterprise client and SAS Analytics Platform server:** When Enterprise Miner is installed in the enterprise client and SAS Analytics Platform server configuration, the enterprise client only needs the Java files that are required to display the client application on the client computer. The enterprise clients connect to the SAS Analytics Platform server, where SAS and Enterprise Miner processing occurs. Enterprise Miner configuration settings are made in the shared SAS Analytics Platform, so there is no need to configure individual Enterprise Miner installations at the client. The client/server configuration enables multiple users to work on the same project, disconnect and reconnect to model training processes, and share mining results packages in a safe and stable environment.

It is a best practice to deploy the SAS Enterprise Miner client using Java Web Start. Java Web Start uses a link in a Web document to provide data when the Enterprise Miner client requests it. Java Web Start automatically downloads the most recent versions of the required files from an application server, and then stores them on the client computer. If the SAS Analytics Platform is configured to run its embedded HTTP server, then the Java Web Start feature is automatically enabled.

You must define SAS Enterprise Miner users in the SAS Management Console. All Enterprise Miner users require permission to use the Enterprise Miner application and the SAS workspace servers where data mining projects are executed.

During SAS installation, a SAS Metadata Server, an object spawner, and a SAS Workspace Server are defined and configured for use by SAS Enterprise Miner. The deployment process also creates a foundation metadata repository and some initial users. For more information, see the SAS 9.2 Intelligence Platform System Administration Guide at <http://support.sas.com/92administration>.

[SAS Enterprise Miner Plug-in for SAS Management Console](#)

- [Overview](#)
- [Server Extensions](#)
- [Configuring SAS Management Console](#)
- [The Projects Folder](#)
 - [Customizing Logical Workspace Servers for Enterprise Miner](#)

[Overview](#)

The SAS Management Console has a plug-in for Enterprise Miner 6.2. This plug-in enables you to browse and customize some of the metadata objects that Enterprise Miner uses. The Enterprise Miner plug-in provides access to attributes of the Logical Workspace server and a server-specific list of projects. The Enterprise Miner plug-in also provides access to the list of mining result packages that have been registered via Enterprise Miner.

[Server Extensions](#)

You can use metadata extensions to Logical Workspace Server definitions to customize the behavior of Enterprise Miner to do the following:

- specify default locations for new projects on a server.
- specify whether users can modify the default location for new projects on a server.
- specify the maximum number of concurrent tasks that are allowed on a particular server. The maximum concurrent tasks setting affects the extent of parallel processing that Enterprise Miner can use in a data mining process flow. You can use the maximum number of concurrent tasks setting to tune your symmetric multiprocessor server environment for optimal performance.
- specify SAS initialization code to be run when a project is opened, or when process flow diagram results or model result packages are generated. SAS initialization code is similar to the Project Start-up Code feature of Enterprise Miner projects, except that SAS initialization code can initialize all SAS sessions for all projects that are associated with a particular server.
- provide an alternate command to launch any SAS/CONNECT sessions that are called for nodes in a data mining process flow.

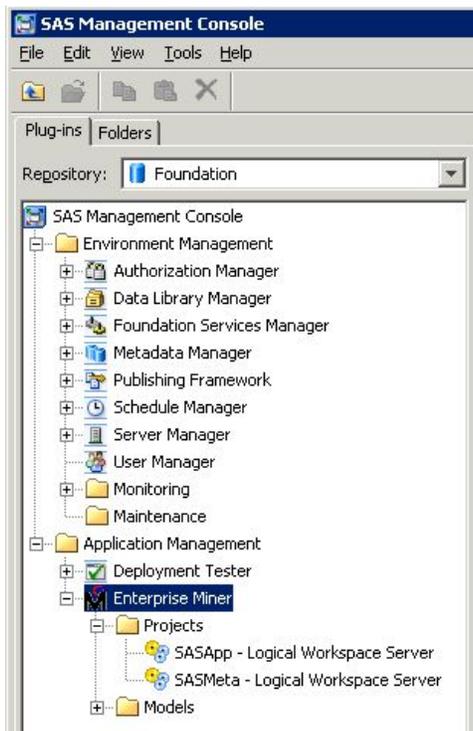
[Configuring SAS Management Console](#)

Before you can use the Enterprise Miner plug-in for SAS Management Console, you must have started an

Enterprise Miner client session at least once. The Enterprise Miner client session initializes the metadata repository with Enterprise Miner metadata. After you successfully start the initial Enterprise Miner client session, you can end the session and start up the SAS Management Console. It is not necessary to define or open any Enterprise Miner projects to initialize the metadata repository.

When you install Enterprise Miner, the Enterprise Miner plug-in is copied to the directory where your SAS Management Console is installed. After you install the Enterprise Miner plug-in, the Enterprise Miner application icon appears in the Application Management section of the left pane of the SAS Management Console window.

In the Application Management section, expand the Enterprise Miner icon to display the Projects and Models folders, as shown in the following display:



[The Projects Folder](#)

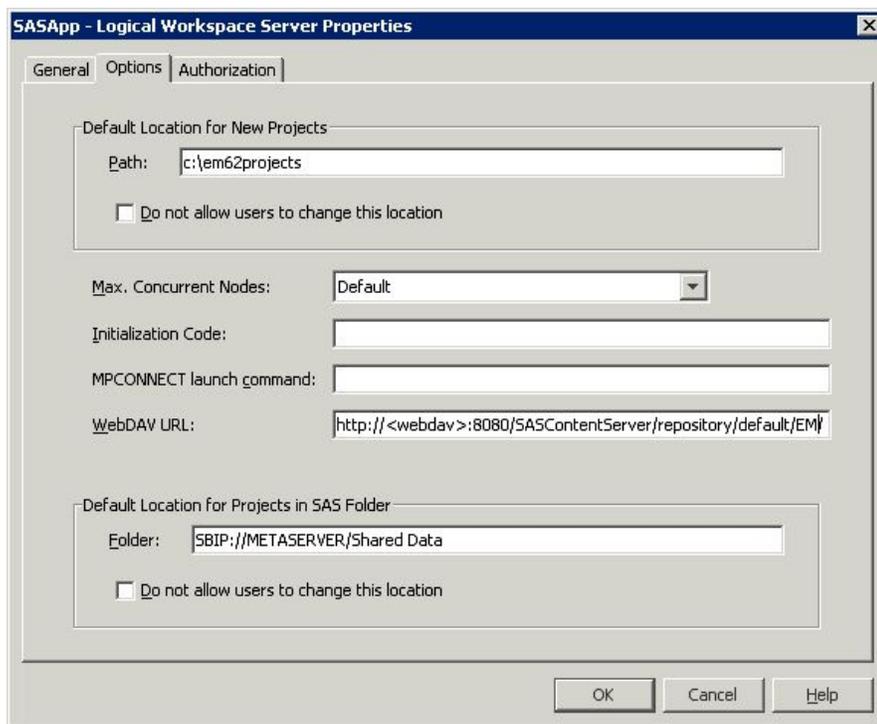
The Enterprise Miner Projects folder contains a list of all of the Enterprise Miner servers. When you select the icon for SASApp - Logical Workspace Server, a list of all the projects created by that server displays in the adjacent pane on the right. Expand the Models folder to access the list of model results packages registered from Enterprise Miner to this server.

You can view and customize Logical Workspace Servers, even if you do not have any registered Enterprise Miner projects. Servers will recognize custom project settings when new projects are created on the servers.

[Customizing Logical Workspace Servers for Enterprise Miner](#)

To customize the properties for an Enterprise Miner server, perform the following steps:

1. Right-click the icon for a server under the Projects folder, and then select Properties from the pop-up menu. The Logical Workspace Server Properties window opens.



2. You can customize the Default Location for New Projects by entering a path. All new Enterprise Miner projects that are created on this server will default to the path that you specify.
3. If you want to prevent users who create projects from changing the default project location, select the *Do not allow users to change this location* check box.
4. In the Max. Concurrent Nodes list, specify the maximum number of concurrently running branches that you want to allow in project process flow diagrams.
5. In the Initialization Code box, enter the path (on this server) to a project start-up file, if you wish to use one. A project start-up file is a text file that contains SAS code that you would like to run when a project is opened or if a process flow diagram is run or if result reports are generated.
6. In the MPCONNECT launch command box, you can enter an alternate command to use when you launch MPCONNECT sessions. Normally it is safe to leave the MPCONNECT launch command box blank. There might be cases where you would like to modify some SAS system defaults for sessions that are used when running process flow diagrams. The following default command is used when this box is left blank:

```
!sascmdv -noobjectserver -nosyntaxcheck -noasynchio
```

This command has the same effect as using the SAS command that was used to launch the SAS workspace session at the time the project was opened.

7. In the WebDAV URL box, you can optionally enter the URL to your WebDAV server. If you specify a WebDAV server, the SAS Enterprise Miner model result packages that you save will be copied to the WebDAV server location when the model packages are registered.

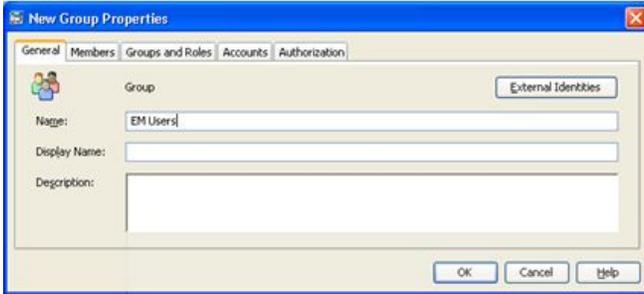
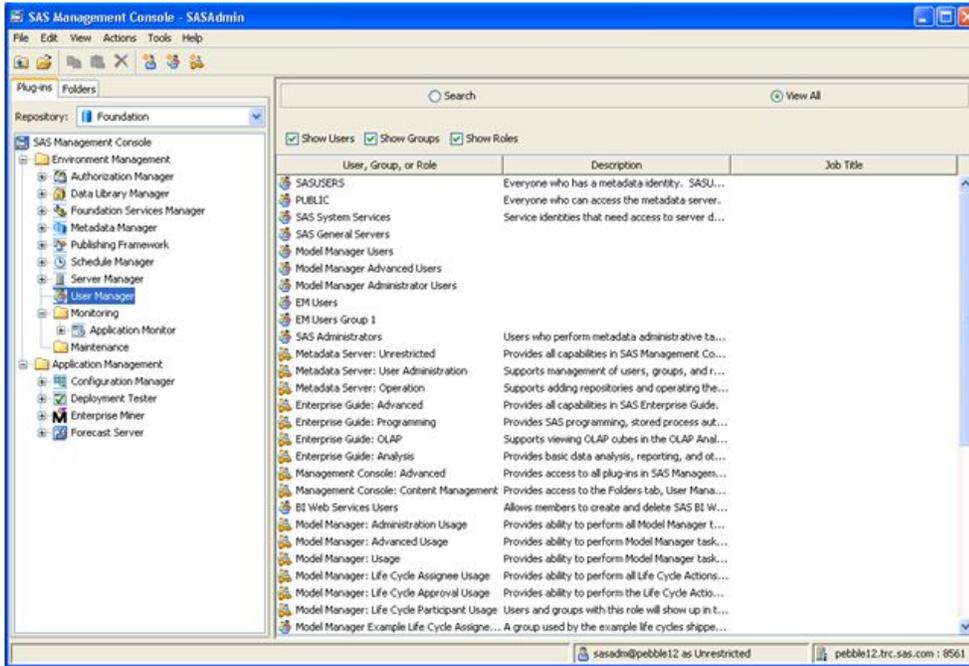
[Adding a New Enterprise Miner User Identity in the SAS 9.2 Management Console](#)

This section provides instructions on how a SAS Administrator can create a new Enterprise Miner identity in the SAS Management Console using domain (or machine name) credentials and a user account that already exists on the machine. The user ID must exist on the operating system before the user ID can be created via SAS Management Console. The user ID and password information on the **Login** tab of the SAS Management Console must exactly match the operating system user ID and password values.

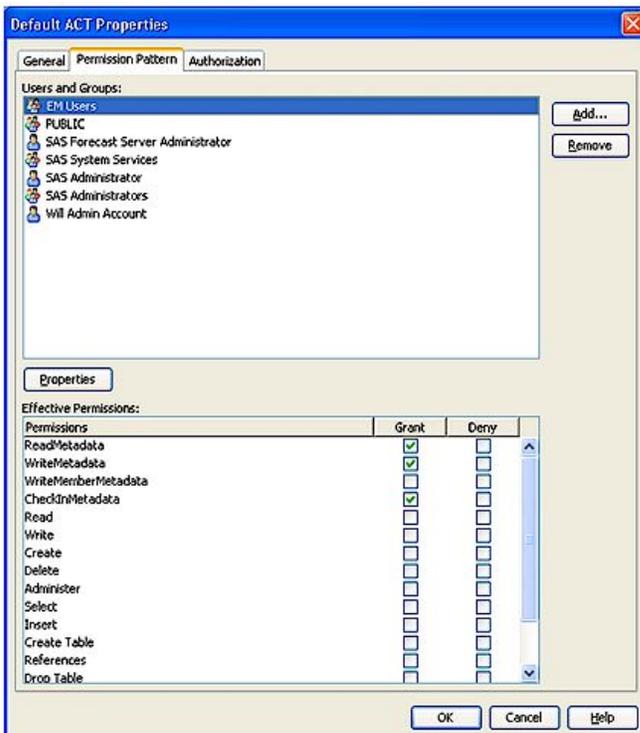
Follow the instructions below to create an identity within the SAS Management Console. After you create the new identity, users will be able to log into Enterprise Miner using these credentials.

1. Log into SAS Management Console as Administrator (sasadm@saspw).
2. Use SAS Management Console to create a group for your Enterprise Miner users. In the SAS Management Console window, right-click the **User Manager** folder and choose **New ➤ Group**. Name your new group

"EM Users" or a name that you prefer.



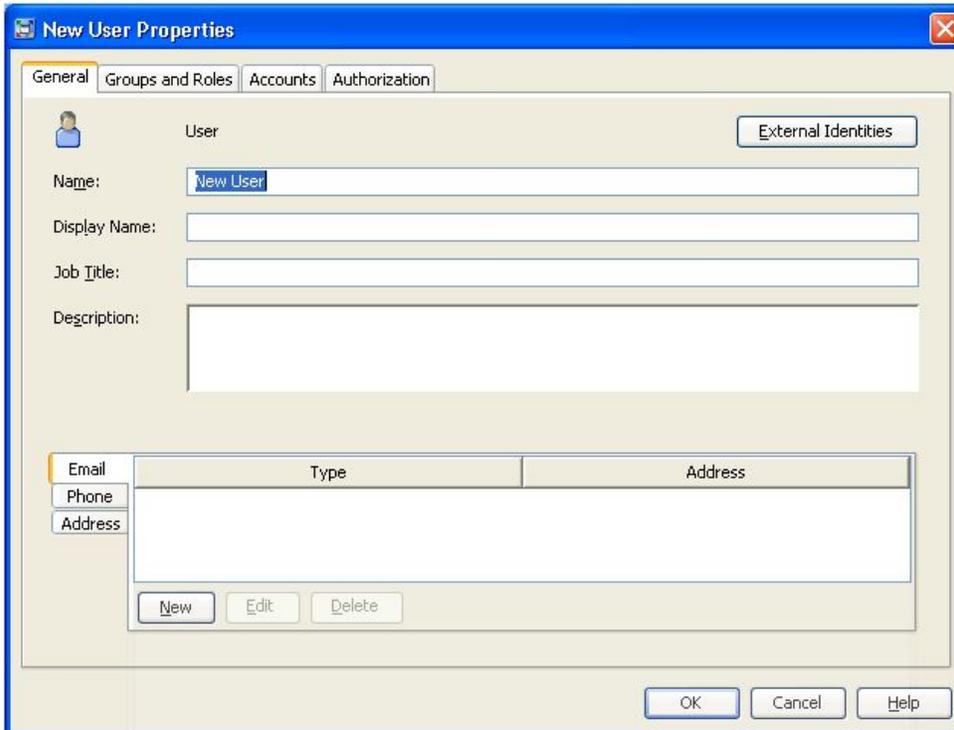
- Now, specify permissions at the default Access Control Template (ACT) level. This is a group level that lets the permissions cascade down to the user level. This means you will not need to specify permissions and authorizations for each individual user since you are managing permissions at the group level. To specify permissions at the ACT level, do the following:



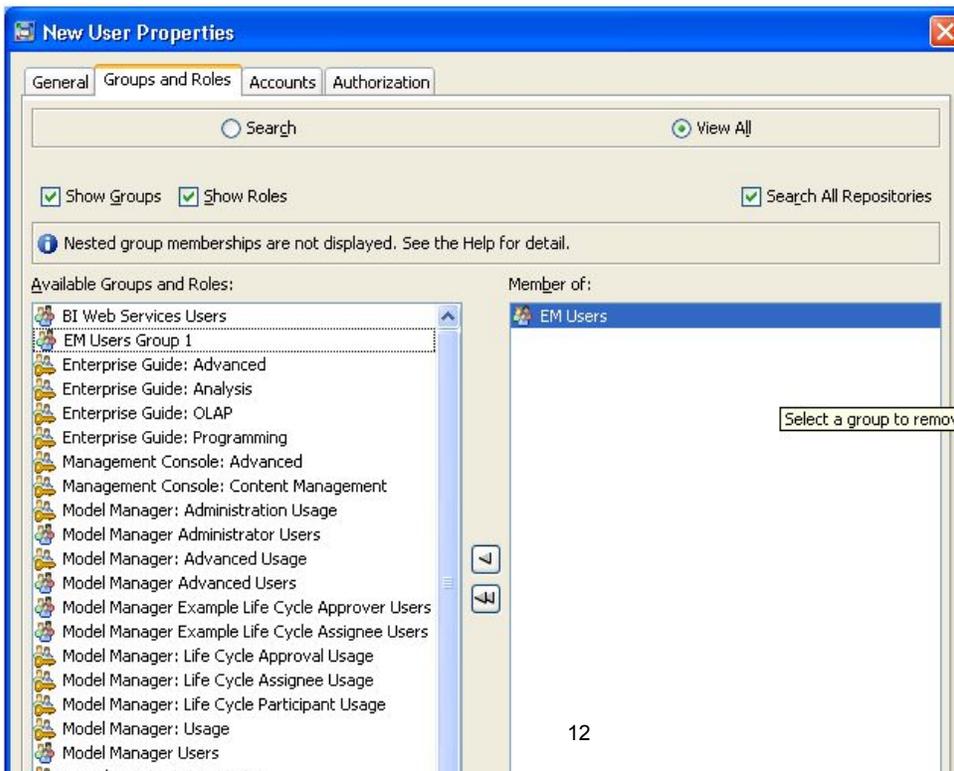
- a. In the SAS Management Console, open the Authorization Manager icon (located just beneath the Environment Management folder) and select **Authorization Manager** ➔ **Access Control Template**.
- b. Right-click on **Default ACT** and choose **Properties**.
- c. Select the **Permission Pattern** tab, click the **Add** button, and move the newly created group from **Available** to **Selected** and click **OK**.
- d. Next, grant the group the following authorizations in the Effective Permissions pane:
 - ReadMetaData
 - CheckInMetaData
 - WriteMetaData

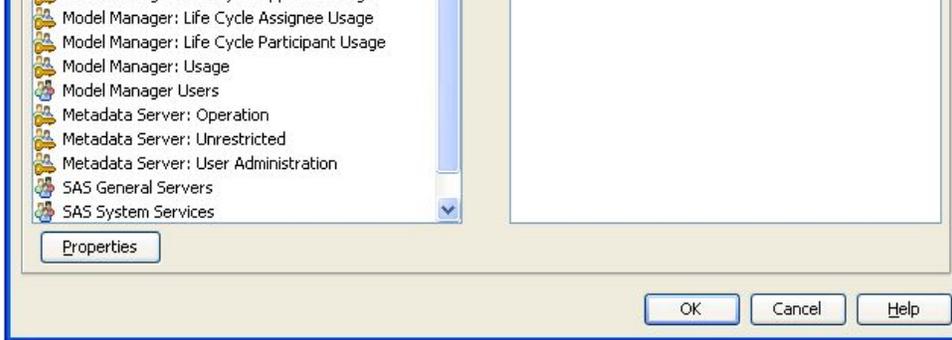
Then click **OK**.

4. Now you are ready to create user identities and to add them to the group. Right-click the User Manager folder, and choose **New** ➔ **User**.
5. In the New User Properties window, enter the full name of the new user in the **Name** field.

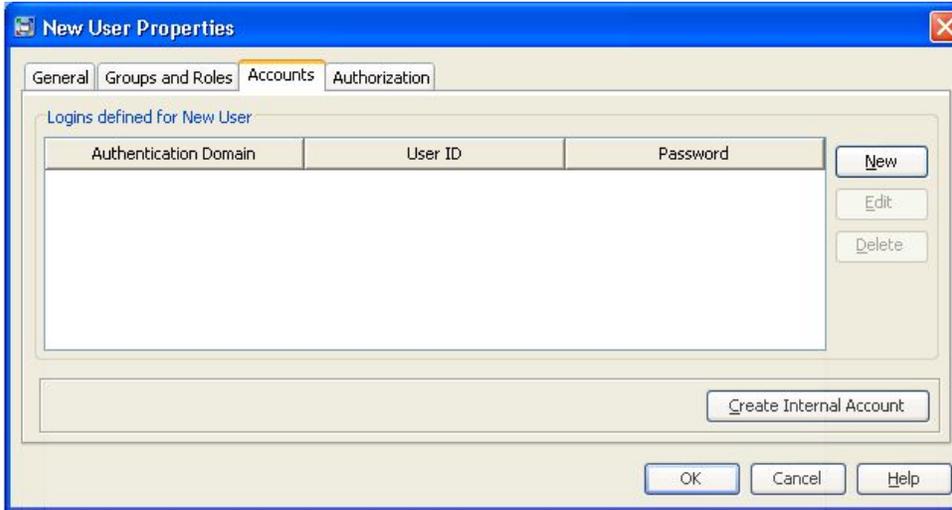


6. Click on the **Groups and Roles** tab, and drag the group that you just created to the Member Of pane and click **OK**.





7. Click on the **Accounts** tab, and then click the **New** button.



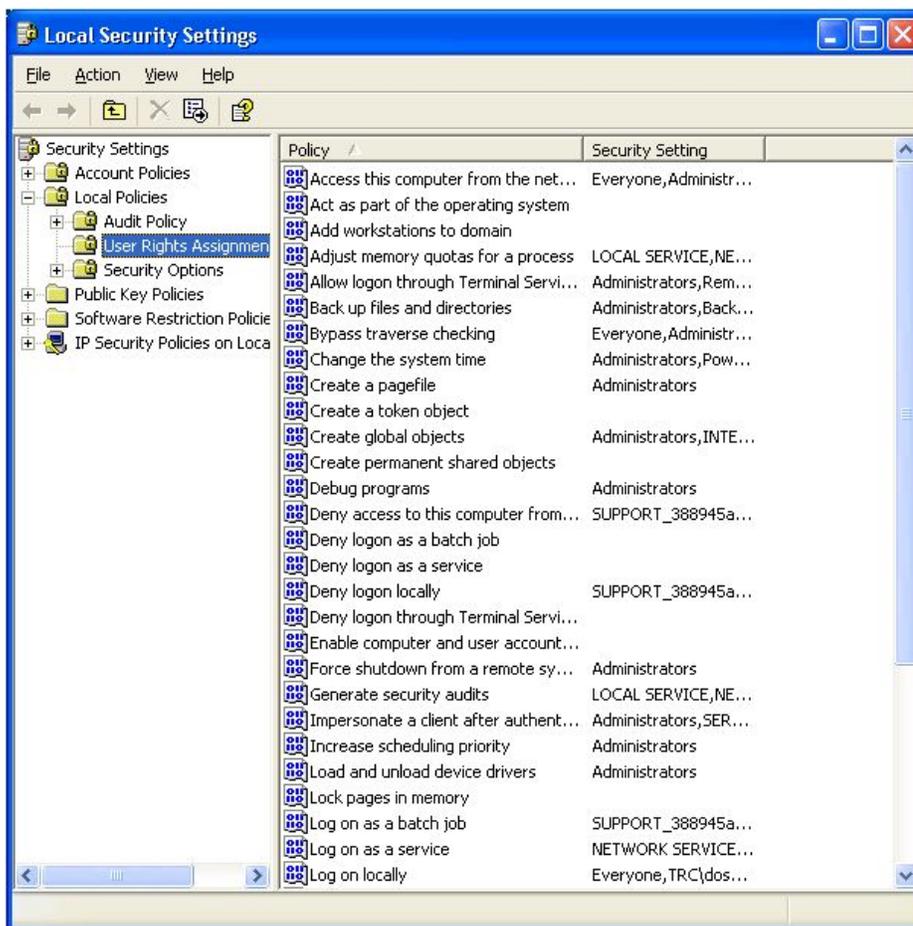
8. Enter the domain credentials and password and then click **OK**. If you are creating the account locally, then use the format of *machine name\id* and then click **OK**.



9. If the ID is not already added to the **Log on as a batch job** privilege, you will need to go through the steps below:

- o Select the following, beginning with your Windows **Start** button:

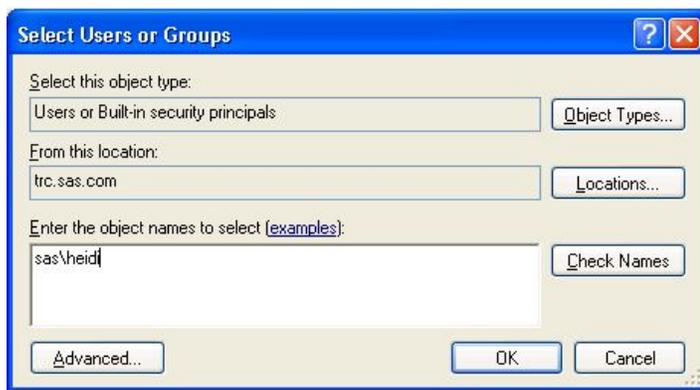
Start ➔ **Settings** ➔ **Control Panel** ➔ **Administrative Tools** ➔ **Local Security Policy** ➔ **Local Policy** ➔ **User Rights Assignments**



- o Right-click on **Log on as a batch job** and choose **Properties**.



- o Click the **Add User or Group** button. The Select Users or Group window opens.



Under **Enter the object names to select...** type in the *domain\ID* (or *computer name\ID*) and click **OK**.

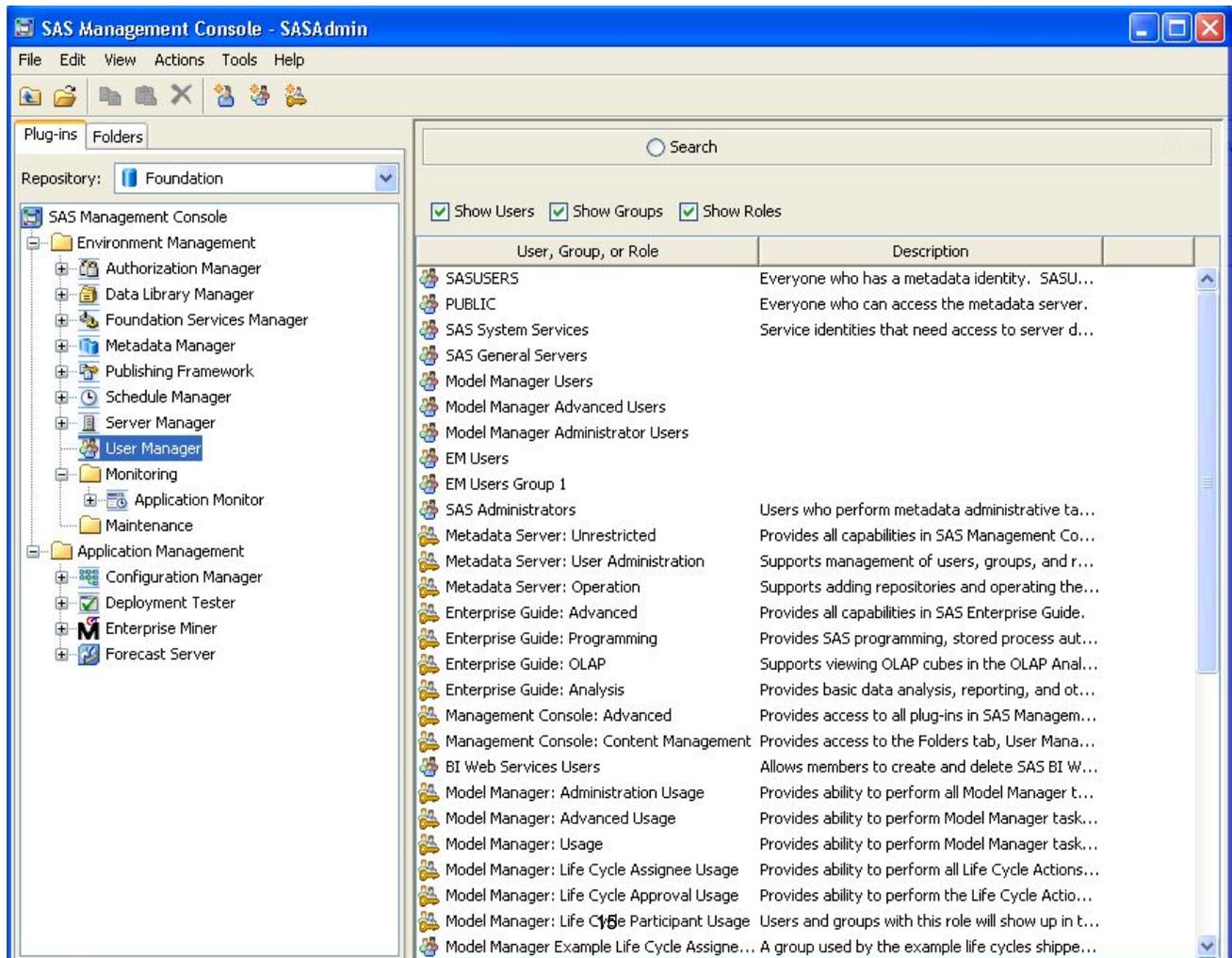
Note: You may also choose to type "Everyone" in the object name field, exactly as shown in the quotation marks. If you do this, everyone who logs into the computer has this right. There is no need to ever do this step again. You can also choose to create an operating system-level group and add this group to the **Log on as batch** privilege. Then, you only need to add members to that group instead of doing this step for each person.

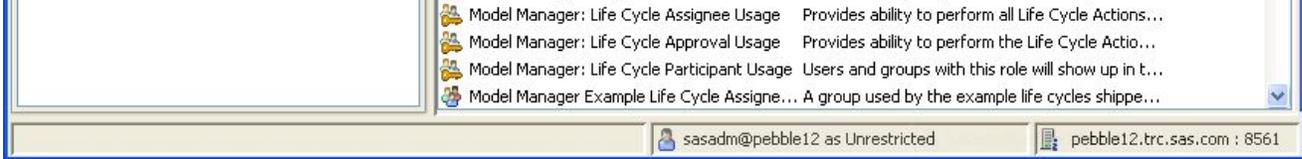
- o The newly created ID should now be able to log into Enterprise Miner.

Creating Different Workspace Servers in SAS Management Console for Enterprise Miner Project Segregation

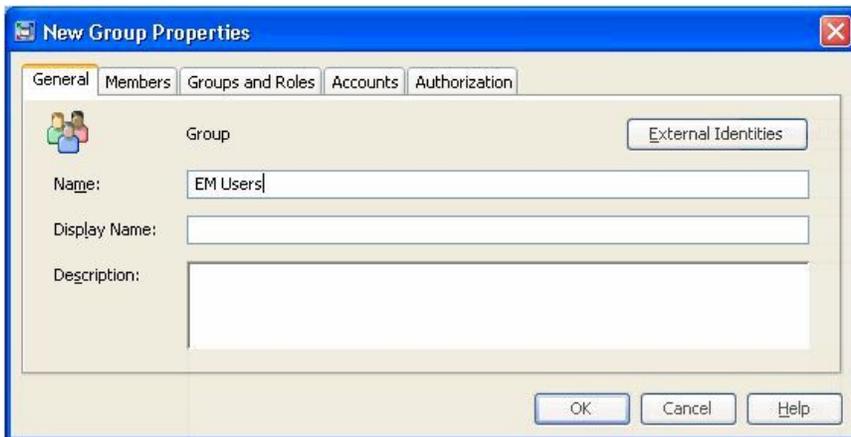
This section demonstrates how to use SAS Management Console to create different workspace servers that you can use to segregate and manage SAS Enterprise Miner projects.

1. Log into SAS Management Console using administrator privileges (sasadm@saspw).
2. Right-click User Manager in the left panel of the SAS Management Console - SASAdmin window and select **New** → **Group**.

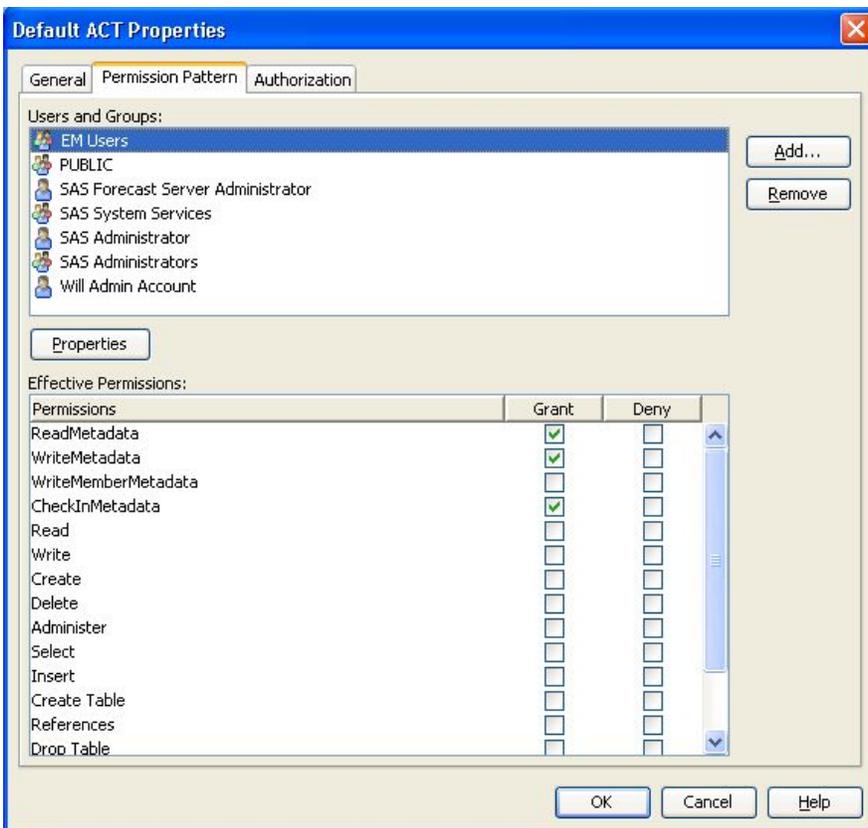




3. Create a new group. This example uses a group named "EM Users".



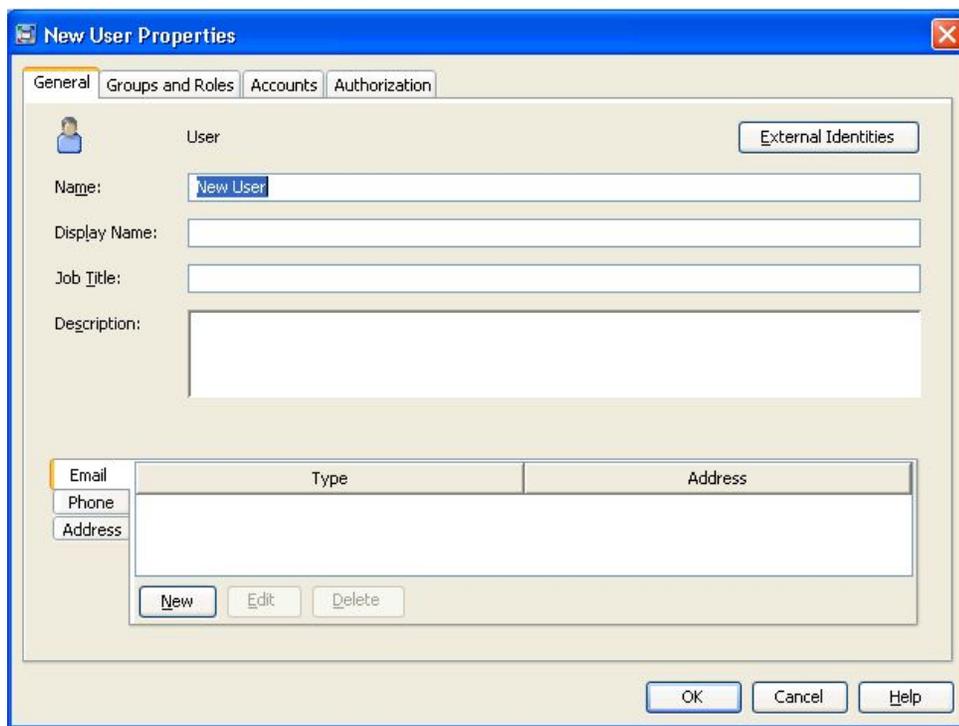
4. Next, you want to set the new group permissions at the Access Control Template (ACT) level. When you use Access Control Template permissions, the permissions cascade down to the user level. You can set permissions at the group level without having to specify authorizations for each user individually. Right-click the Authorization Manager in the left panel of the SAS Management Console window, and select **Authorization Manager** ➔ **Access Control Template** ➔ **Default ACT**. The Default ACT Properties window opens.



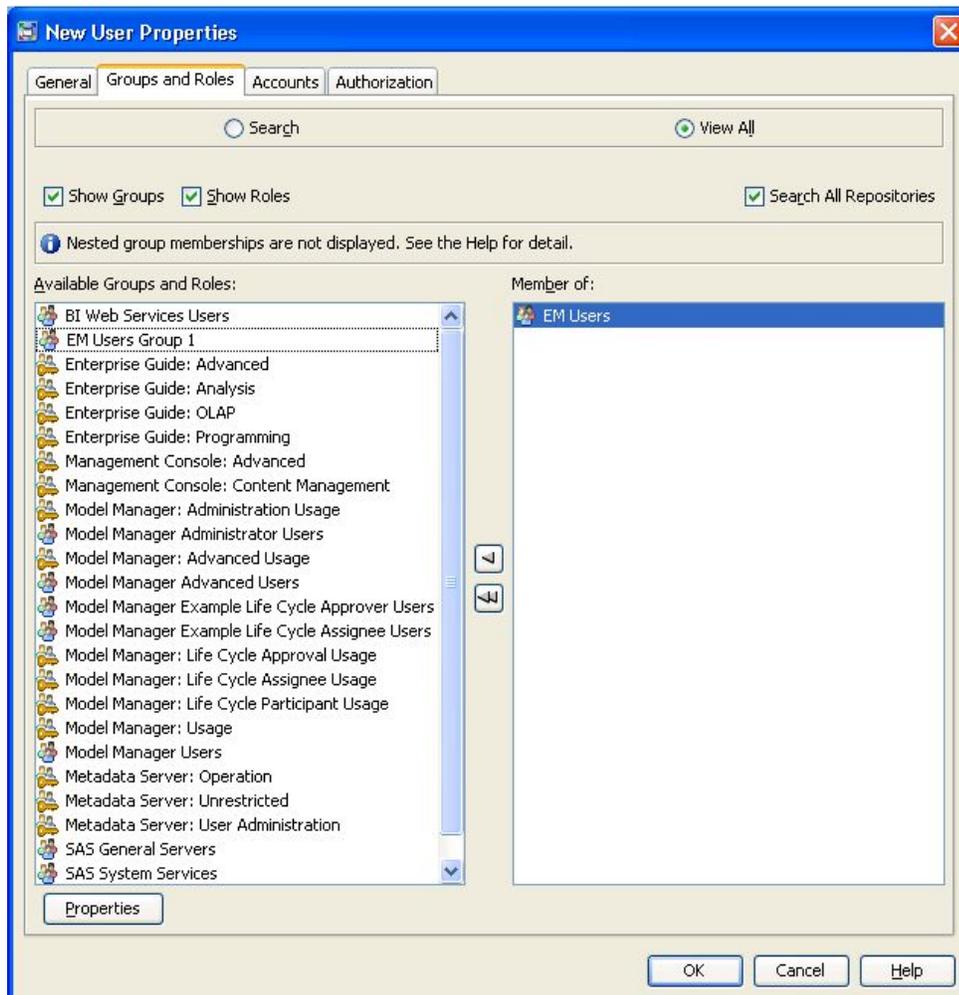
5. Select the **Permission Pattern** tab, highlight your new group, and then click **Add**. Move the group from **Available** to **Selected** and then click **OK**. Then grant the group effective permissions for **ReadMetadata**, **WriteMetadata**, and **CheckInMetadata**, and click **OK**.

6. Now you can create user identities and add them to the group, or you can add existing user identities to the group. Right-click on the User Manager icon in the left panel of the SAS Management Console window, and select **New** ➔ **User**.

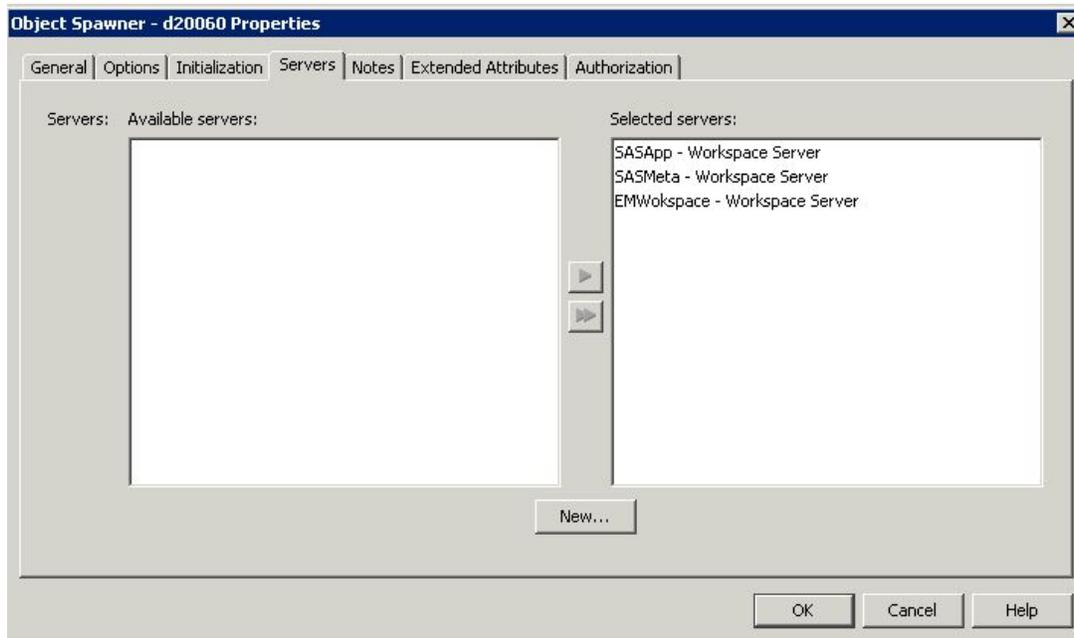
Note: For existing user identities, select **Add Members to the EM Users group**.



7. In the New User Properties window, enter the user's name in the **Name** field. Then click on the **Groups and Roles** tab, and drag the newly created group from the **Available Groups and Roles** list to the **Member Of** list and click **OK**.

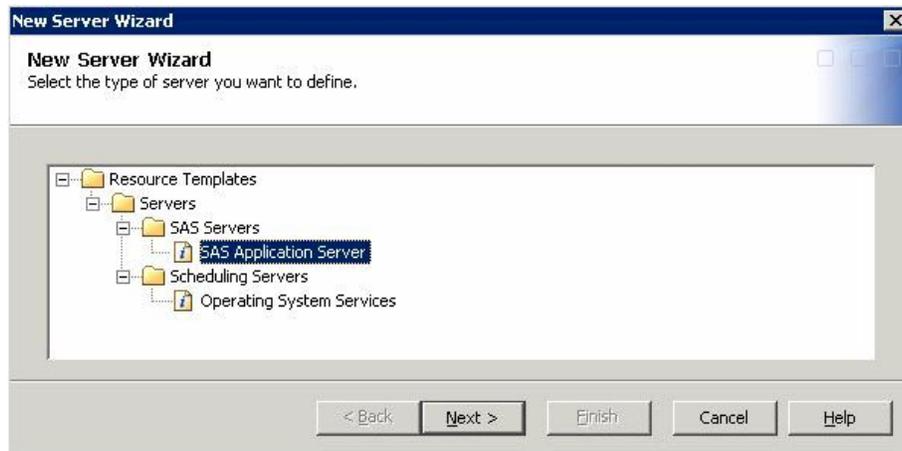


8. In the SAS Management Console window, expand the Server Manager icon, and then right click the Object Spawner icon and choose **Properties**. The Object Spawner Properties window opens.

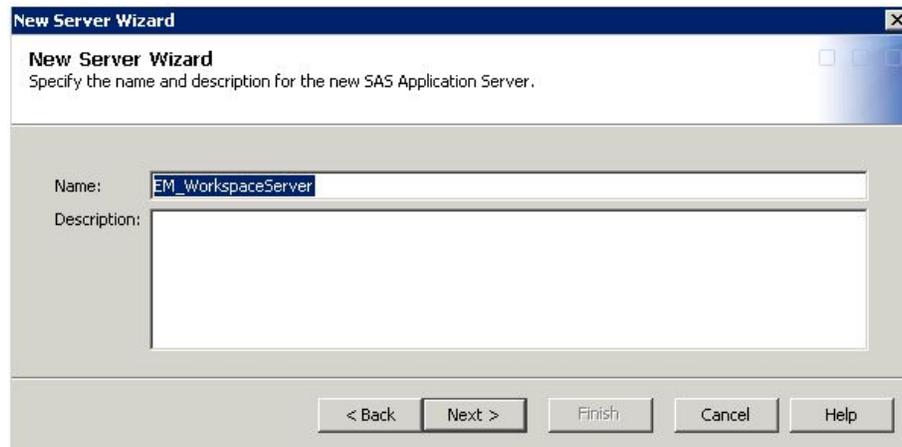


In the Object Spawner window, select the **Servers** tab and click **New**.

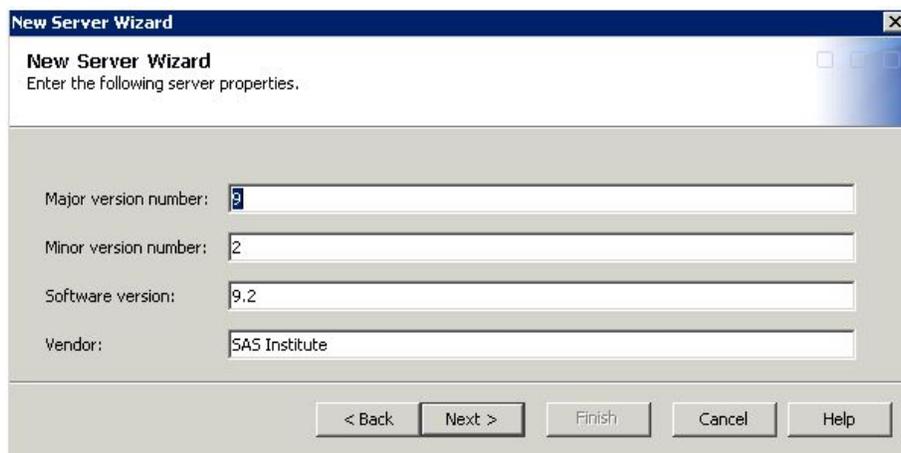
9. You use the New Server Wizard to create a new SAS Application Server. In the New Server Wizard, select SAS Application Server and click **Next**.



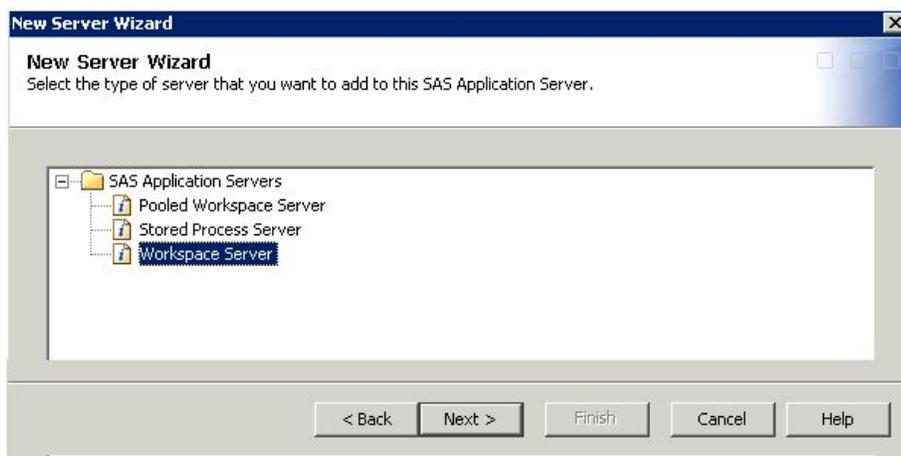
10. Assign a name to your new SAS Application Server. This example uses EM_WorkspaceServer. Click **Next**.



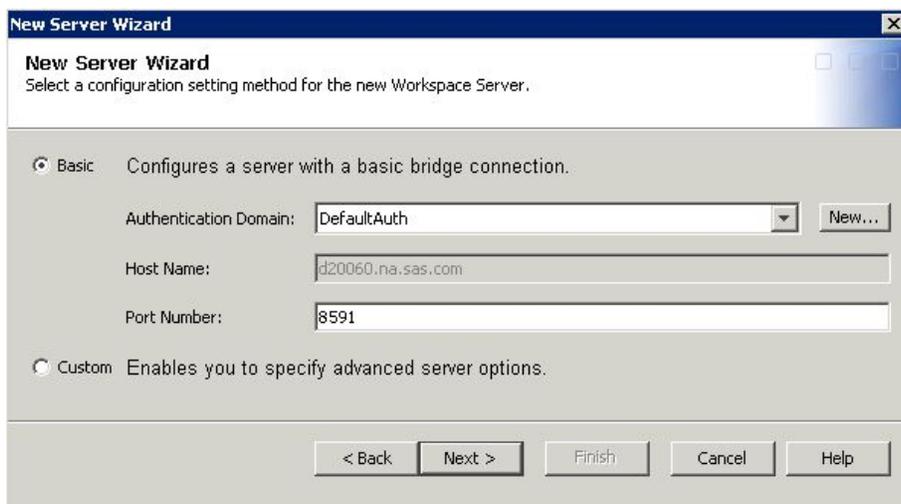
11. In the Server Properties, accept the default settings, and click **Next**.



12. Specify the type of server that you want to add to this SAS Application Server. Select **Workspace Server** and click **Next**.

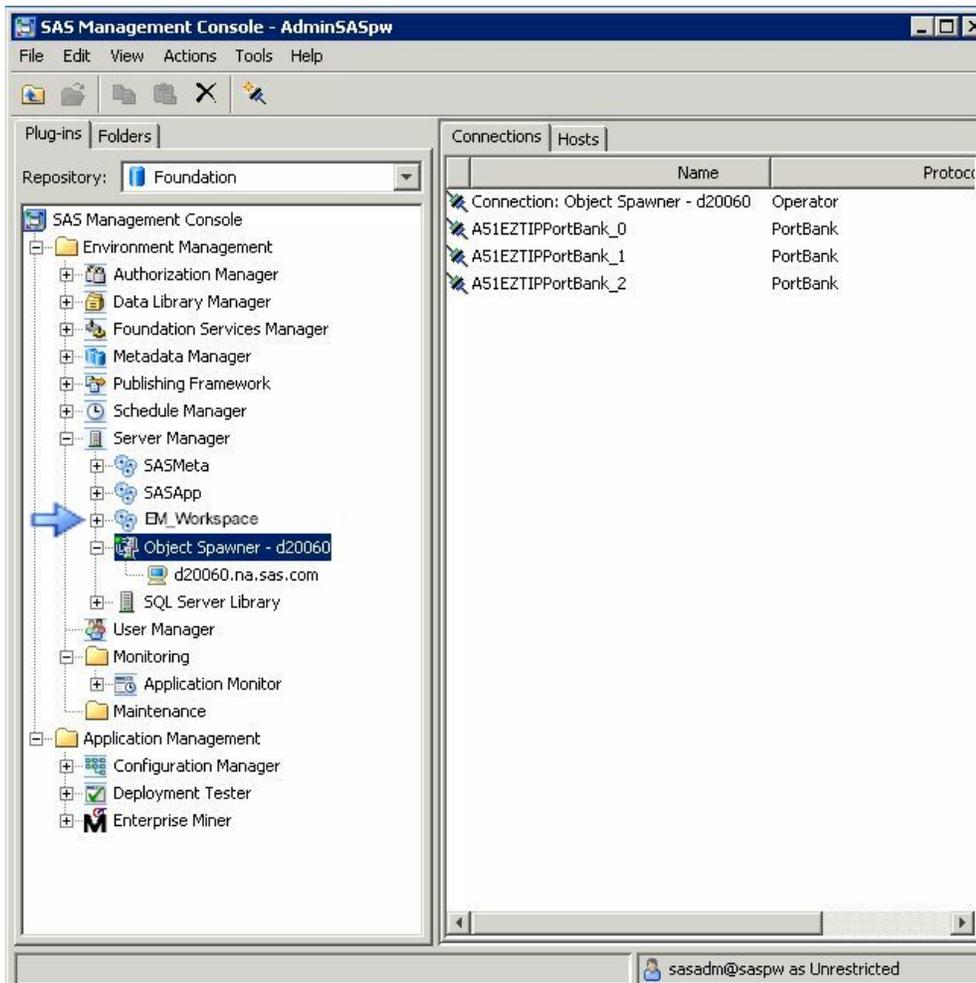


13. Select a configuration setting method for your new Workspace Server and click **Next**, and then **Finish**.

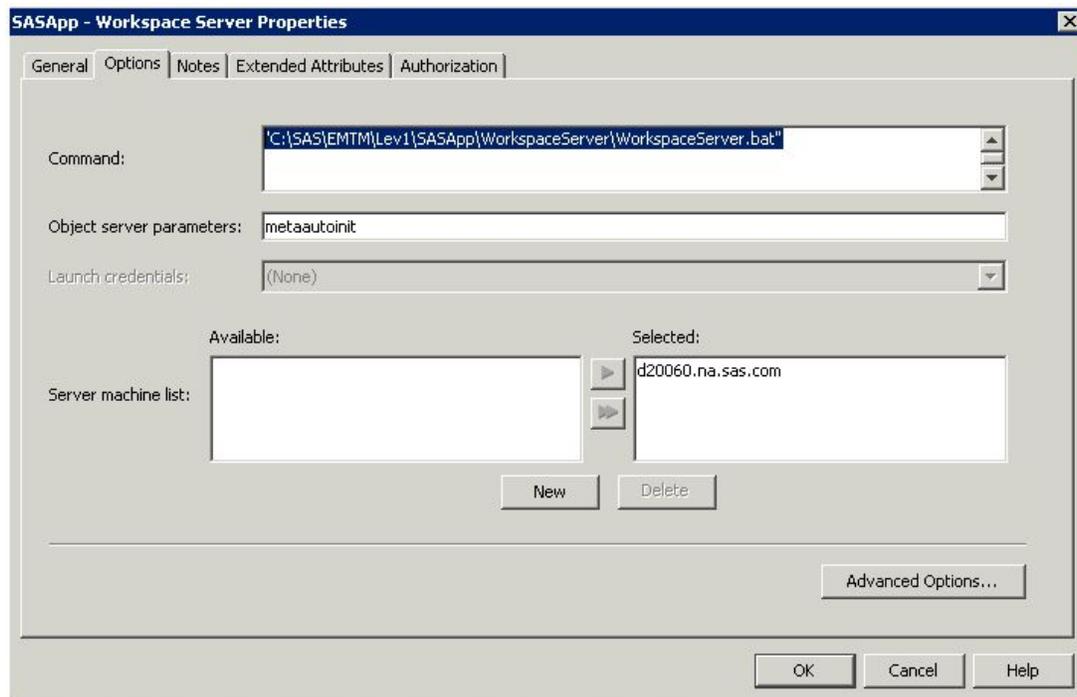


Note: Your host name and port number assignments will vary from those shown in this example.

14. After you have created your new server, return to the SAS Management Console main window. Right-click the Server Manager icon, and choose **Refresh**. Your newly created server should appear in the expanded Server Manager list.



- Now you need to copy the launch command for your SASMain workspace to your new server. In the SAS Management Console window, expand the Server Manager icon as follows: **Server Manager** → **SASMeta** → **SASMeta - Logical Workspace Server**. Then, right-click **SASMeta - Logical Workspace Server** and choose **Properties**. In the Workspace Server Properties window that opens, click the **Options** tab.

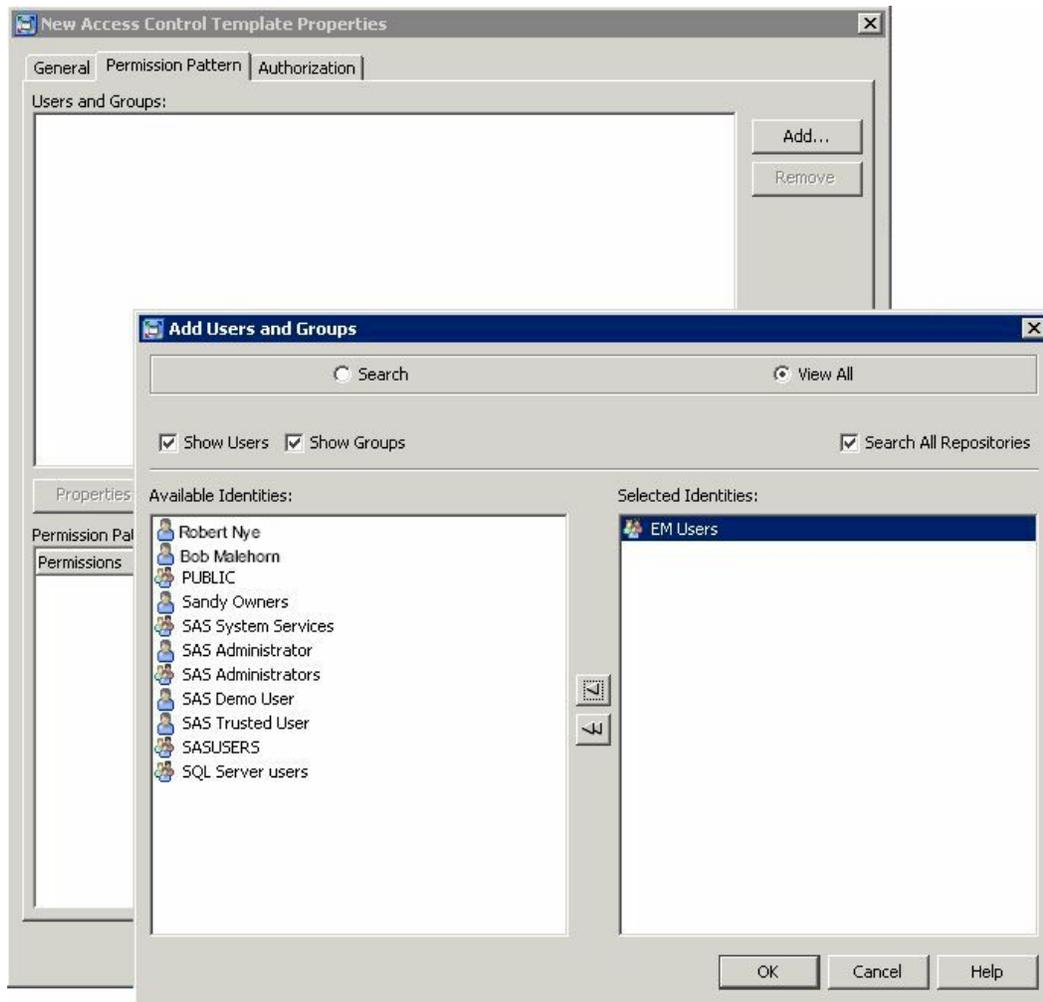


The Command field contains the path to your server launch command. Copy this command, and then paste the command in the **Options** tab of the Properties window for your new server. Your server launch command path might vary from the example above if your installation did not use all of the default SAS installation paths.

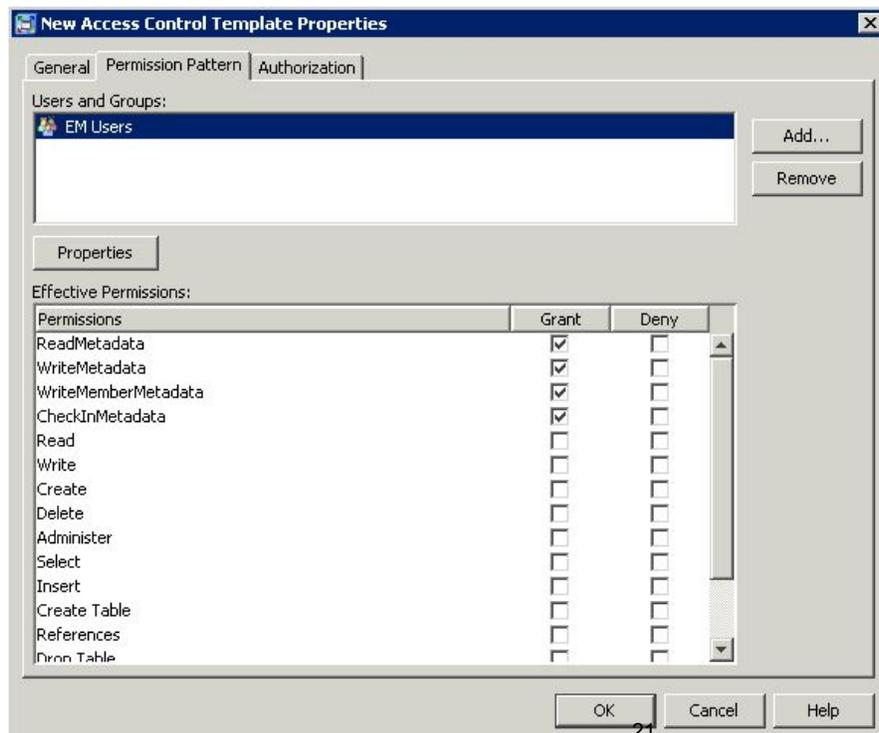
- Now, define the Access Control Templates for your new server. In the SAS Management Console window,

expand the Authorization Manager icon: **Authorization Manager** ➔ **Access Control Templates**. Right-click on **Access Control Templates** and select **New Access Control Template**.

- In the **General** tab of the New Access Control Template window that opens, name your template. Then select the **Permission Pattern** tab, click **Add**, and then add the EM Users group in the Add Users and Groups window and click **OK**.

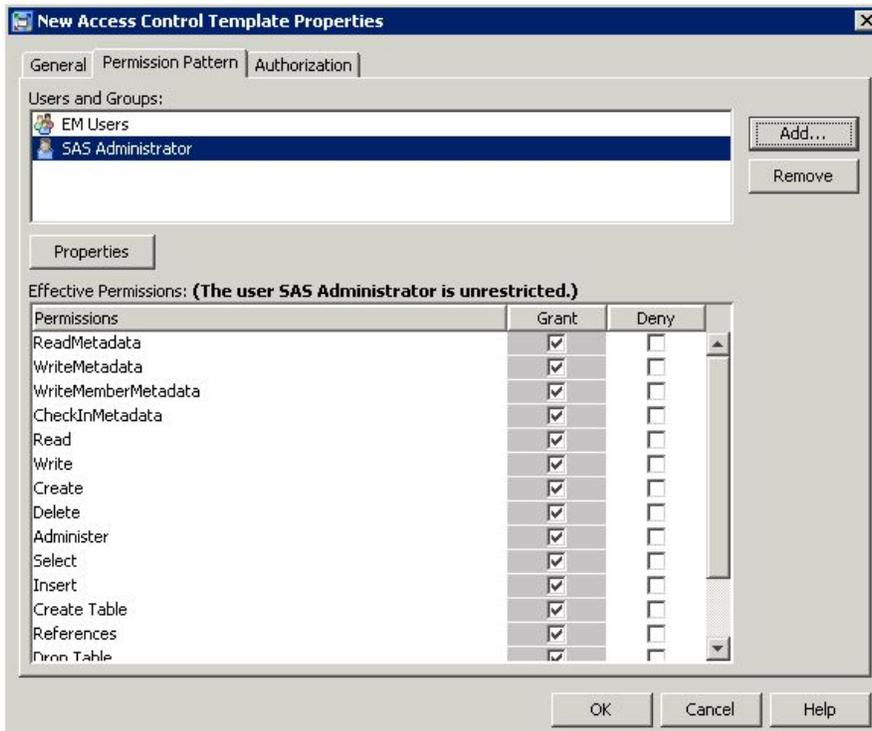


- Highlight **EM Users** in the New Access Control Template Properties window and ensure that the group has the following permissions granted in the Effective Permissions area.

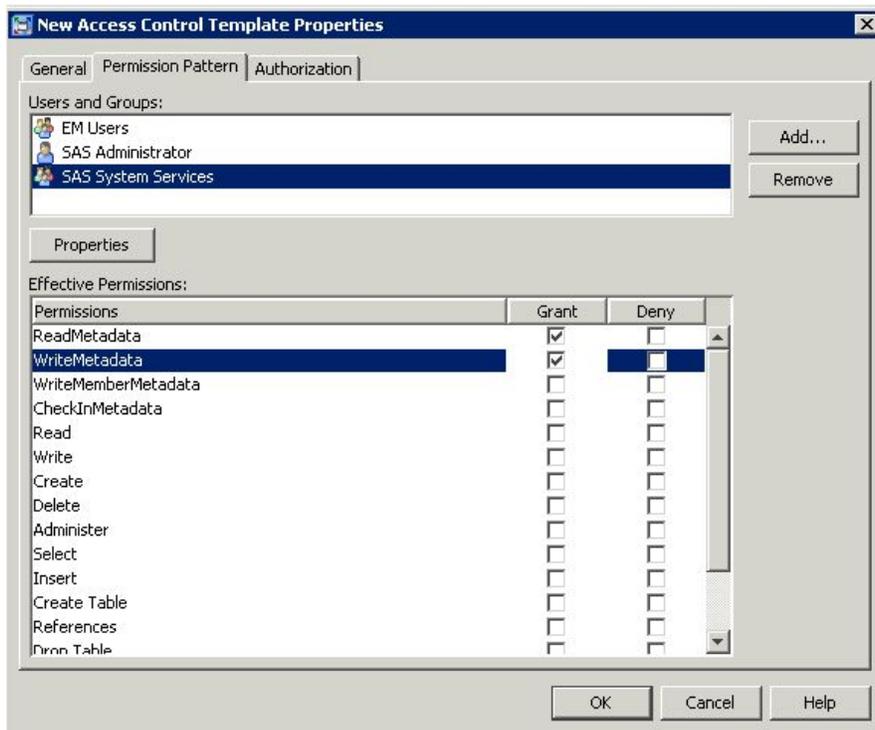


- o ReadMetadata
- o WriteMetadata
- o WriteMemberMetadata
- o CheckInMetadata

19. Click **Add** and then add user **SAS Administrator** to the Users and Groups list, and grant ALL rights.

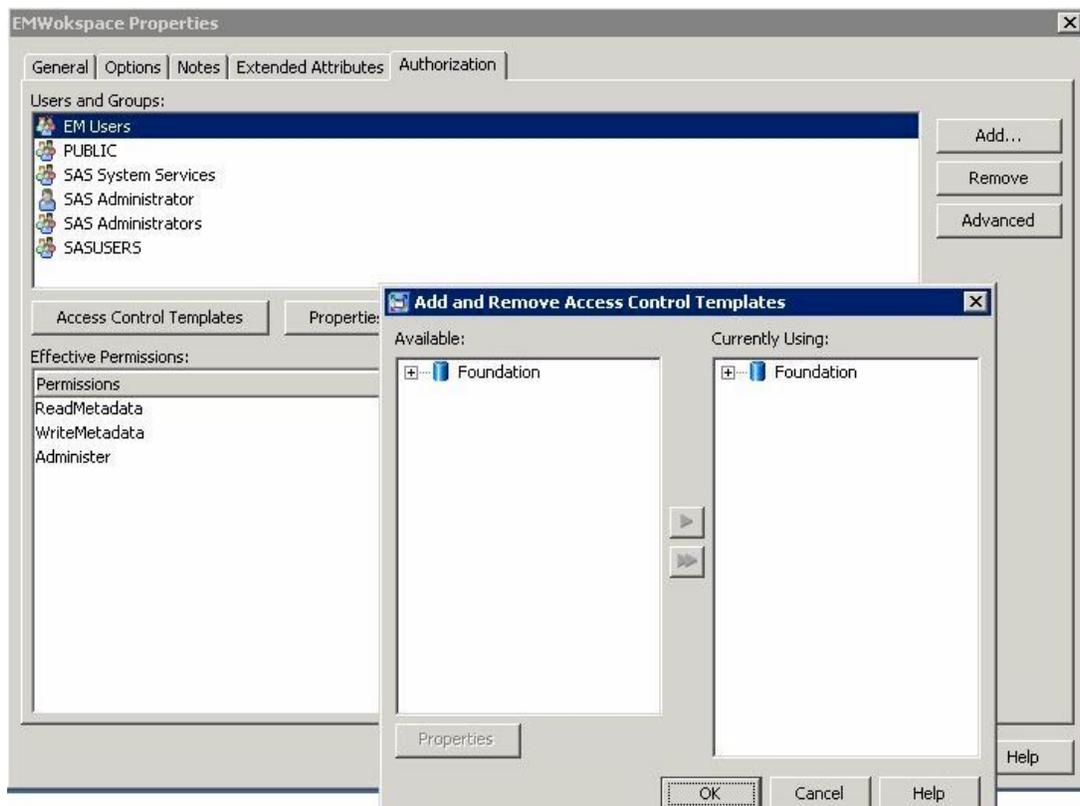


20. Click **Add** and then add group **SAS System Services** to the Users and Groups list, and grant effective permissions for ReadMetadata and WriteMetadata:

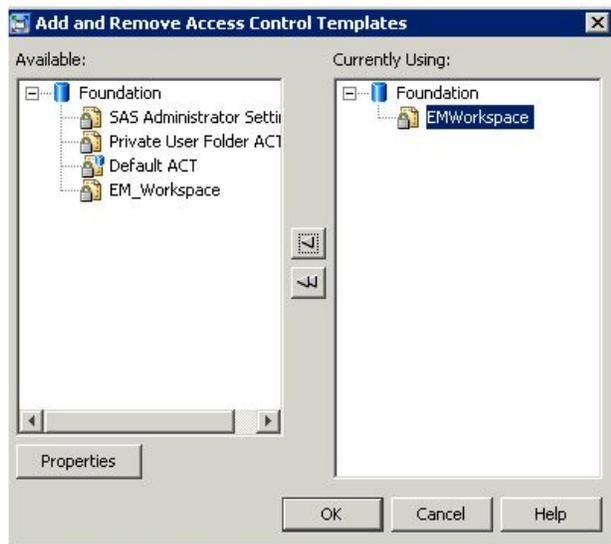


Then click **OK** to save the access control template for your new server.

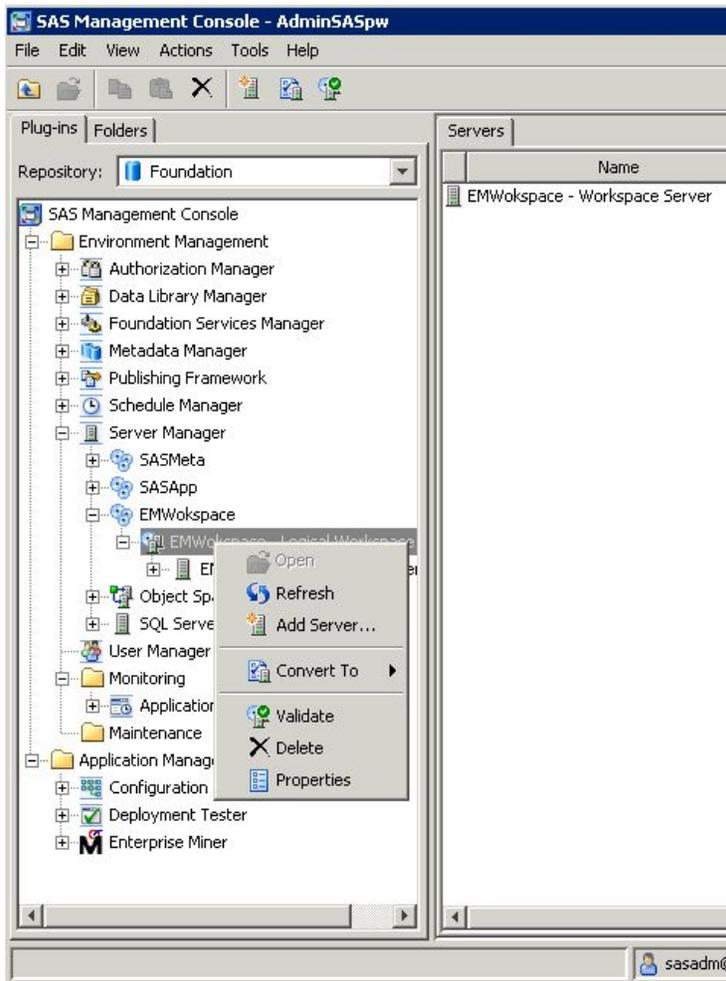
21. Now you can define server access permissions to your workspace servers by applying the access control template that you created. In the SAS Management Console window, right-click your newly created workspace server at the highest level, and select **Properties**. The properties window for your new workspace server opens. Select the **Authorization** tab and highlight your EM Users group, then click the **Access Control Templates** button to open the Add and Remove Access Control Templates window:



22. In the Add and Remove Access Control Templates window, expand **Foundation** in the left pane. Then, drag the new template **EMWorkspace** from the expanded list into right pane and click **OK**.



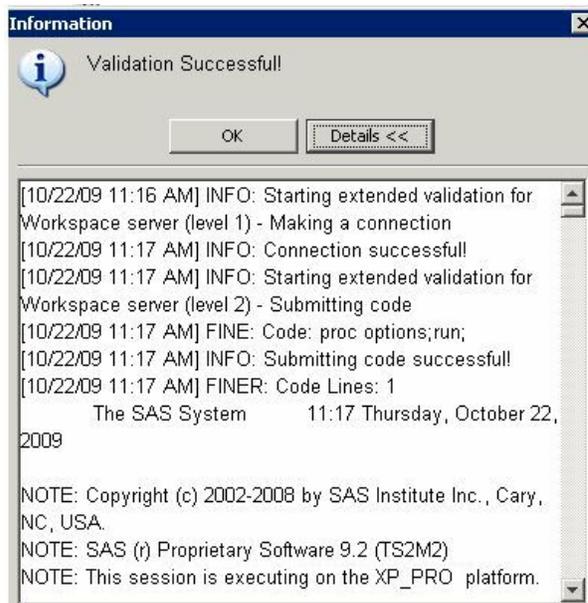
23. Test your server connection by doing the following:
 - o Stop and restart the Object Spawner.
 - o In the SAS Management Console window, expand the Server Manager icon, then expand **EMWorkspace**, right-click on **EMWorkspace - Logical Workstation**, and select **Validate**:



- o Enter a valid user ID and password for one of the EM Users in the Log On window that opens:



You should receive notification of a successful validation similar to the following:



24. Users that belong to the **EM Users** group should now be able to log on to SAS Enterprise Miner and to select the new Workspace Server that you just created for their projects.
-

SAS Enterprise Miner 6.2 Client Start-up Properties

SAS Enterprise Miner provides you with the ability to make additional configuration changes to your Enterprise Miner client installation. You can customize SAS Enterprise Miner client properties by editing the Enterprise Miner application configuration file, `app.config`. The `app.config` file is typically located in the Enterprise Miner configuration directory. SAS Enterprise Miner configuration directories are normally installed to a location resembling the following:

```
C:\SAS\EMTM\Levl\AnalyticsPlatform\apps\EnterpriseMiner
```

Property names that begin with the prefix "application" are properties that are automatically configured by the SAS Deployment Wizard. Do not change properties that begin with the prefix "application."

Property names that begin with the prefix "webstart" affect only the SAS Enterprise Miner client during client sessions that are initiated via Java Web Start. Power Enterprise Miner users that work with complex projects might need additional memory. You can provide such users with access to additional memory by increasing the size of the two memory properties shown below. You can increase the `max.heap.size` property to a value such as 1024m. However, in most cases, default user memory allocations should suffice.

```
webstart.client.initial.heap.size=128m
webstart.client.max.heap.size=512m
```

Property names that begin with the prefix "em" control how the SAS Enterprise Miner client starts up in a single-user workstation configuration. In a single-user workstation configuration, all SAS components are installed on the end user's system, and there are no connections to remote services. Initial values for properties with the "em" prefix are set depending on the type of Enterprise Miner installation configuration that was specified by the SAS Deployment Wizard.

The SAS Deployment Wizard offers three basic deployment patterns: Enterprise Client, Desktop Client, and Classroom Client. You can modify these deployment patterns to suit your custom needs. For example, if you have an Enterprise Client deployment, but would like to avoid the Enterprise Client login dialog box, can do so by specifying the following property values:

```
#workstation properties

em.workstation.only=Y

em.local.server=SASMain - Logical Workspace Server

em.login.prompt=N

em.login.remember.user=Y

em.login.remember.password=Y

em.project.access=localfilesystem
```

When you edit any SAS Enterprise Miner configuration file, be sure that you create a backup copy of the original file before saving your new version. If your configuration file becomes unusable, you might need to run the SAS Deployment Wizard again to create a new valid configuration.

SAS Enterprise Miner 6.2 Projects

When a SAS Enterprise Miner user creates a new project in the software, a corresponding project object is created in the SAS Metadata server. The project object provides essential project information to the SAS Workspace server. The project objects specify the locations of Enterprise Miner project data sets, catalogs, and files, as well as determining where data mining calculations are executed.

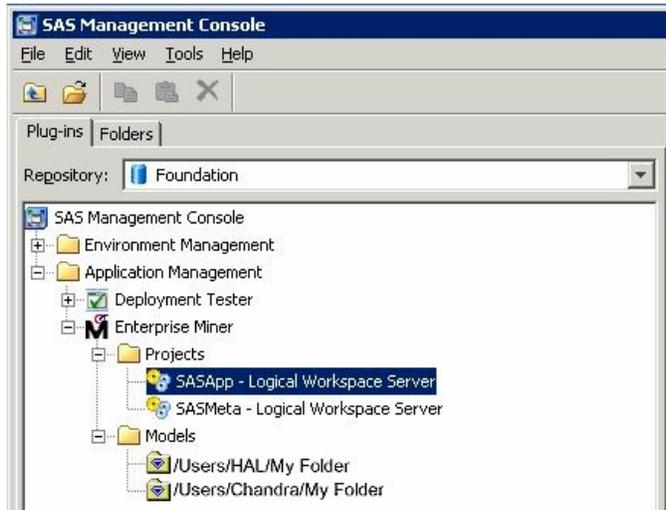
Metadata objects such as file and catalog locations are defined in the SAS Folders of your Enterprise Miner project metadata. You can use the SAS Management Console to view and change permissions and properties of your project metadata objects. The default location for new metadata objects is My Folder.

SAS Enterprise Miner data mining projects that were migrated from a SAS 9.1.3 installation will be located in the folder named EnterpriseMinerProjects. The EnterpriseMinerProjects folder is located inside the Shared

Data folder in the SAS Metadata server.

You can view Enterprise Miner project metadata objects via the **Plug-ins** tab of the SAS Management Console window. To view all project objects, expand the following folders:

Application Management ➤ Enterprise Miner ➤ Projects



SAS Enterprise Miner 6.2 Models

When an Enterprise Miner user registers a model package, a corresponding new model object is created in the SAS Metadata. Model package objects are also known as Mining Result objects. Mining Result objects contain the following data:

- SAS score code needed for deployment
- lists of input and output variables, types, and formats
- meta information such as model type and model target variables
- name of the algorithm that was used
- name of the data mining analyst
- time of creation

Model package objects are defined in the SAS Folders structures of the SAS Metadata server. You can use the SAS Management Console to change permissions and properties of Model Package objects. The default location for new objects in SAS Management Console is My Folder.

Models that are migrated from a SAS 9.1.3 installation are stored in a folder called EnterpriseMinerModels. The EnterpriseMinerModels folder is located inside the Shared Data folder in the SAS Metadata server. You can move your model objects to other shared or private locations in the SAS Folders structure.

The Application Management folder in the SAS Management Console contains a list of all metadata objects that are associated with a particular model. To view the list of SAS Enterprise Miner metadata objects, expand the following folders in the **Plug-ins** tab of SAS Management Console:

Application Management ➤ Enterprise Miner ➤ Models



When a model package is registered, a corresponding SAS model package file can be stored in a WebDAV location if desired. SAS provides WebDAV storage through the SAS Content Server product. SAS model package files contain complete data mining results sets, including process flow diagram configuration, model property configuration settings, generated reports, data mining summary tables, SAS log and output listings, graphic plots, score code, and related scoring information.

[Allocating Libraries for SAS Enterprise Miner 6.2](#)

- [Overview: Allocating Libraries](#)
 - [Allocate Libraries via a SAS Autoexec File](#)
 - [Allocate Libraries via Server Initialization Code](#)
 - [Allocate Libraries via Project Start Code](#)
 - [Allocate Libraries via SAS Management Console](#)
- [ERROR: Data Set LIBREF.TABLENAME Does Not Exist](#)

[Overview: Allocating Libraries](#)

In SAS Enterprise Miner 6.2, there are several places where LIBNAME statements (or other initialization code) can be specified. The library allocations can be specified in these locations:

- [SAS Autoexec Files](#)
- [Server Initialization Code](#)
- [Project Start-Up Code](#)
- [The SAS Management Console Library Manager Plug-In](#)

The general form of the LIBNAME statement is as follows:

```
LIBNAME libref "path";
```

For example, you can specify the following statement:

```
LIBNAME MYDATA "d:\EMdata\testdata";
```

(Windows path examples are given, but the same principles apply to UNIX systems.)

[Allocate Libraries via a SAS Autoexec File](#)

If LIBNAME statements are specified in an autoexec.sas file that resides in the SAS root path, then they execute by default for all SAS processes except those that explicitly specify an autoexec override. You can specify the path to a specific autoexec.sas file by adding the option to the workspace server's SAS launch command or to any sasv9.cfg file:

```
-autoexec "[full path]"
```

Note: You cannot use a mapped drive specification to indicate the path to an autoexec.sas file.

In most installations, Enterprise Miner uses the configuration file that is located here:

```
C:\SAS\EMiner\Levl\SASApp\sasv9.cfg
```

In this example, **EMiner** is the installed plan name and might vary from site to site. If you do not designate a plan name, then the default path will be as follows:

```
C:\SAS\Config\Levl\SASApp\sasv9.cfg
```

The sasv9.cfg file in this directory includes the sasv9.cfg file that is located in the SAS root directory:

```
C:\Program Files\SAS\SASFoundation\9.2\sasv9.cfg
```

The sasv9.cfg file in the SAS root directory points to the last configuration file located in the **nls\en** subdirectory:

```
C:\Program Files\SAS\SASFoundation\9.2\nls\en\sasv9.cfg
```

[Allocate Libraries via Server Initialization Code](#)

You can also specify LIBNAME statements that are specifically used with SAS Enterprise Miner. These statements are unavailable to other users of the workspace server.

To execute LIBNAME statements for every SAS Enterprise Miner project on a server, follow the instructions in the Enterprise Miner Help in the Installation and Configuration section, "Preparing SAS Enterprise Miner for Use." Also see the Enterprise Miner Help on "Customizing SAS Enterprise Miner Metadata," which explains how to use the Enterprise Miner plug-in to SAS Management Console to specify the path to the file that contains the server initialization code.

[Allocate Libraries via Project Start-up Code](#)

You can use Enterprise Miner project start-up code to issue LIBNAME statements for individual SAS Enterprise Miner 6.2 projects. To modify the start-up code for an Enterprise Miner project, open the project in Enterprise Miner, go to the Navigation panel, and select the project name at the top of the navigation tree. With the project highlighted in the Navigation panel, go to the Properties panel, locate the Start-Up Code property, and click the  button in the Value column. Enter the LIBNAME statement in the Start-up code window and click **OK** to save your new project's start-up code. You can also choose to execute the start-up code immediately by clicking on the **Run Now** button. A **Log** tab is available so that you can view the SAS log after executing your start-up code.

[Allocate Libraries via SAS Management Console](#)

Enterprise Miner data libraries that are used frequently can be allocated for use with SAS Enterprise Miner 6.2 using SAS Management Console.

First, you must define the library for the SAS Enterprise Miner input data set:

1. Open SAS Management Console.
2. Under the Data Library Manager plug-in, right-click on the **Libraries** folder and select **New Library**.
3. Select the appropriate engine. If the SAS data set is located on the SAS Workspace Server, your engine should be the SAS base engine. Select **SAS Base Library** and click **Next**.
4. Type the name of your library and click **Next**. Be sure to check the **Location** and be sure that the selected value is appropriate. The **Location** specifies the metadata folder that contains your library definition.
5. Select an available server from the list on the left and click the right arrow. This will move the selected server into the adjacent Selected servers pane. Click **Next**.

Note: If the SASMeta server appears in the list, do *not* select it as your server.

6. Enter a libref for the library in the **LIBREF** field. The libref must be 8 characters or fewer.
7. Click **New** and enter the name of the directory where the library is located.

Note: This directory must be accessible to the SAS Workspace Server.

8. Click **Advanced Options**, select the **Library is pre-assigned** check box, and click **OK**.
9. Click **Next** and review your entries. Text similar to this should be displayed:

```
Library:           My Enterprise Miner data
Libref:           emdata
Location:         /Shared Data
Assigned to SAS Servers: SASApp
Libref:           MyData
Engine:           BASE
Path Specification: c:\yourdata <specify correct path to data>
Library is pre-assigned: Yes
```

If this looks correct, click **Finish** and then **OK**.

Next, you must grant read permission for the metadata in your new library:

1. In SAS Management Console, click on the Data Library Manager icon.
2. Expand the **Libraries** folder.
3. Right-click the SAS library that you just created and select **Properties** from the pop-up menu.
4. In the Library Properties window, go to the **Authorization** tab and select the appropriate user or user group who needs access to the library (for example, **SASUSER**).
5. Select the check box in the **Grant** column for the Read permission row, and ensure that the ReadMetadata check box is selected.

ERROR: Data Set LIBREF.TABLENAME Does Not Exist

In SAS Enterprise Miner 6.2, nodes that follow a SAS Code node or custom node in a process flow diagram can produce an error that indicates that the data set that a node attempted to reference does not exist. You might get this error even when you are able to successfully create the data source and can explore the data set in your session. In SAS Enterprise Miner 6.2, each node in a process flow diagram spawns a new SAS session. The currently executing node does not have access to libraries that were allocated via the SAS Program Editor or a predecessor SAS Code node. In order for SAS libraries to be available to all tools and nodes in SAS Enterprise Miner 6.2, the LIBNAME statements must be specified in a location that is executed for each spawned session, such as in the [project start-up code](#), the [server initialization code](#), or [SAS Management Console](#).

SAS Enterprise Miner 6.2 PMML Support

- [PMML Overview](#)
- [Enabling PMML Generation](#)
- [Enterprise Miner 6.2 Nodes that Support PMML](#)
- [PMML Document Structures](#)
- [SAS Extensions to PMML](#)
- [SAS Formats and Informats in PMML](#)
 - [Example of SAS Format to PMML Conversion](#)
- [SAS User-Defined Formats in PMML](#)
- [Text Normalization in PMML](#)
- [Decision Processing in PMML](#)
- [Scoring A PMML Document](#)

PMML Overview

PMML (Predictive Modeling Markup Language) is an XML-based standard for representing data mining results. The PMML standard was developed by the Data Mining Group (DMG), an independent group of software vendors. PMML is designed to enable the sharing and deployment of data mining results between vendor applications and across data management systems. The Data Mining Group released PMML Version 2.1 in March, 2003. More information about DMG can be found at

<http://www.dmg.org>.

Enterprise Miner produces PMML files for selected data mining functions. Enterprise Miner 6.2 PMML is updated to full compatibility with DMG Version 3.1. The Enterprise Miner PMML files use significant extensions in order to support the data types, transformations, and model definitions that SAS requires.

Enabling PMML Generation

By default, Enterprise Miner nodes do not generate PMML. To enable the creation of PMML code, you must submit the following macro in the Enterprise Miner editor, or in your start-up or project code.

```
%let EM_PMML=Y;
```

Enterprise Miner 6.2 Nodes That Support PMML

The table below outlines the Enterprise Miner nodes that generate PMML output.

Enterprise Miner Node Name	Scoring Functions	Generates SAS Code?	Generates PMML?
Association	Probability, Classification, and Decision Assignment	Yes	Yes
Cluster	Segment and Distance Assignment	Yes	Yes
Regression	Probability, Classification, and Decision Assignment	Yes	Yes
Dmine Regression	Probability, Classification, and Decision Assignment	Yes	Yes
Decision Tree	Probability, Classification, Decision Assignment, and Leaf Assignment	Yes	Yes
Neural Network	Probability, Classification, and Decision Assignment	Yes	Yes
AutoNeural	Probability, Classification, and Decision Assignment	Yes	Yes

PMML Document Structures

Each PMML document has a fixed structure with the following components:

- [Header](#)
- [Data Dictionary](#)
- [Transformation Dictionary](#)
- [Model](#)
 - [Mining Schema](#)
 - [Output Section](#)
 - [Local Transformations](#)
 - [Target Section](#)
 - [Model Details](#)

Details on individual sections follow with example PMML code snippets. The code snippet examples are from different models and are not intended as continuity examples.

Header

The PMML header contains timestamp and other system and application information.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
```

```
<PMML version="2.1">
  <Header copyright="Copyright(c) 2002 SAS Institute Inc., Cary, NC, USA.
All Rights Reserved.">
  <Application name="SAS(r)" version="9.13"/>
  <Timestamp>2007-10-07 22:48:30</Timestamp>
```

Data Dictionary

The Data Dictionary contains information on variables in the Data Source. The scoring engine ignores attributes that begin with the prefix "x-SAS".

```
<DataDictionary numberOfFields="20">
  <DataField name="x" optype="continuous" dataType="double"/>
  <DataField name="time" optype="continuous" dataType="timeSeconds"
x-SASinformat="TIME."/>
  <DataField name="y" optype="continuous" dataType="double"/>
  <DataField name="c" optype="categorical" dataType="string"
x-SASformat="$CHAR."/>
  <DataField name="n" optype="categorical" dataType="double"
x-SASformat="BEST12."/>
```

Transformation Dictionary

The Transformation Dictionary contains information on any SAS user-defined formats in use. The logic behind the format is captured as a function definition in generic PMML terms. Note that the PMML is intended for non-proprietary representation.

```
<TransformationDictionary>
  <DefineFunction name="COLOR" optype="ordinal" x-SASuserformat="COLOR">
  <ParameterField name="AnyInput" optype="continuous"/>
  <Discretize field="AnyInput" defaultValue="Purple">
  <DiscretizeBin binValue="Red">
  <Interval closure="closedClosed" leftMargin="1" rightMargin="1"/>
  </DiscretizeBin>
  <DiscretizeBin binValue="Green">
  <Interval closure="closedClosed" leftMargin="2" rightMargin="2"/>
```

Model

The rest of the PMML document contains various aspects of the data mining model as described in the following sections.

```
<RegressionModel functionName="regression"
targetFieldName="y" normalizationMethod="none">
```

Mining Schema

Mining Schema contains information on variable roles in the model, missing value handling, and so on.

```
<MiningSchema>
  <MiningField name="x" usageType="active" optype="continuous"/>
  <MiningField name="color" usageType="active" optype="categorical"/>
  <MiningField name="y" usageType="predicted" optype="continuous"/>
  <MiningField name="customer" usageType="group" optype="categorical"
outliers="asIs" missingValueTreatment="asIs"/>
```

Output Section

Output section contains information on computed fields such as errors and residuals.

```
<Output>
  <OutputField name="P_y" displayName="Predicted: y" optype="continuous"
dataType="double" field="y" feature="predictedValue"/>
  <OutputField name="E_y" displayName="Error Function:
y" optype="continuous"
dataType="double" field="y" feature="predictedValue"/>
```

Local Transformation

Local Transformation contains information on application of formats, normalizations, and encodings.

```
<DerivedField name="n+" optype="categorical">
  <!-- SAS Format: 12.0 -->
  <Apply function="SAS-EM-String-Normalize">
    <Constant>12</Constant>
    <Apply function="SAS-FORMAT-BESTw">
      <Constant>12</Constant>
      <FieldRef field="n"/>
    </Apply>
  </Apply>

<DerivedField name="_1_0" optype="continuous">
  <MapValues outputColumn="To">
    <FieldColumnPair field="n+" column="From"/>
    <InlineTable>
      <row>
        <From>1</From>
        <To>1</To>
      </row>
      <row>
        <From>2</From>
        <To>0</To>
      </row>
    </InlineTable>
  </MapValues>
</DerivedField>
```

Target Section

Target Section enlists possible target or predicted values and any priors.

```
<Targets>
  <Target field="species" optype="categorical">
    <TargetValue value="VIRGINICA" rawDataValue="virginica "
      priorProbability="0.3333333333333333"/>
    <TargetValue value="VERSICOLOR" rawDataValue="versicolor"
      priorProbability="0.3333333333333333"/>
    <TargetValue value="SETOSA" rawDataValue="setosa "
      priorProbability="0.3333333333333333"/>
  </Targets>
```

Model Details

Model Details on the actual model follow. For example, here is a snippet from a regression model showing intercepts and coefficients.

```
<RegressionTable intercept="6.4">
  <NumericPredictor name="_0_0" coefficient="-1.7033934213238E-15"/>
  <NumericPredictor name="_0_1" coefficient="7.1465186199382E-16"/>
  <NumericPredictor name="_1_0" coefficient="1.6"/>
  <NumericPredictor name="_1_1" coefficient="-0.4"/>
</RegressionTable>
```

Here is a code snippet from a tree model showing a split:

```
<Node id="1" score="7.54230769230769" recordCount="130">
  <True/>
  <Node id="2" score="5.33333333333333" recordCount="60">
    <CompoundPredicate booleanOperator="surrogate">
      <CompoundPredicate booleanOperator="or">
        <SimplePredicate field="color+-" operator="equal" value="1"/>
        <SimplePredicate field="color+-" operator="equal" value="2"/>
      </CompoundPredicate>
    </CompoundPredicate>
  </Node>
  <False/>
</Node>
<Node id="5" score="3.225" recordCount="10">
```

The following general extensions were made to PMML 3.1 in order to support SAS models. Contact SAS technical support for detailed documentation on PMML schemas needed to develop PMML interpreters and scoring engines.

- **SAS Informats**
 - Match data between database systems and SAS systems
 - Datetime, Date, Time, Char, Numeric
- **SAS Formats**
 - W, Best, Hour, Downname, Monname, Year, etc.
- **User-Defined Formats**
 - Map Raw Char to Formatted Char
 - Map Raw Numeric to Formatted Char
- **SAS Functions**
 - Uppcase(), Trim(), Left(), Cosine()
- **Modeling**
 - Add lift to associations
 - General class variable dummy encoding
 - Regression interaction terms
 - Regression link functions
 - Neural Radial Basis Function - width and altitude
 - Tree multi-way splits
 - Cluster missing value handling
 - Output field definitions
 - Probability
 - Classification
 - Residual
 - Error

SAS Formats and Informats in PMML

SAS system informats are used to map data values into the SAS environment. This mapping is encoded into PMML as data type parameters in the data dictionary.

SAS has proprietary representation for date and time values. Dates are represented as the number of days since January 1, 1960. Time is represented as the number of seconds since midnight. DATETIME stamps are represented as the number of seconds since midnight of January 1, 1960.

When date and time data is brought into SAS from a DBMS, SAS assigns a date/time informat to the specific variables and converts them for SAS internal representation. Subsequent modeling on such data imported into SAS works with the converted internal values. To enable accurate scoring of these models on new data, a similar conversion must be applied.

Three new data types have been introduced in PMML to represent date and time values:

SAS Informat	PMML Datatype
date	dateDaysSince[1960]
time	timeSeconds
datetime	dateTimeSecondsSince[1960]

A PMML scoring engine must convert the incoming data from the DBMS value to the SAS internal values based on the actual date or time values.

The SAS system offers many formats. Enterprise Miner models class variables based on formatted values. In the absence of a specific format, numeric categorical variables are formatted using the default BEST12 format. Character categorical variables are formatted with the CHAR format.

Formats in PMML are handled as a function application with names of the type SAS-FORMAT-*formatname*. Formats in PMML are usually seen in the local transformation section of the PMML document. The scoring engine implements the SAS-FORMAT-functions.

The following SAS system formats are currently supported in PMML:

SAS System Formats	PMML Function Names
--------------------	---------------------

BEST	SAS-FORMAT-BESTw
\$CHAR	SAS-FORMAT-\$CHARw
DATE	SAS-FORMAT-DATEp
DATETIME	SAS-FORMAT-DATETIMEp
DOWNNAME	SAS-FORMAT-DOWNNAMEw
HOUR	SAS-FORMAT-HOURp
MONTH	SAS-FORMAT-MONTHp
MONNAME	SAS-FORMAT-MONNAMEw
MONYY	SAS-FORMAT-MONYYw
QTR	SAS-FORMAT-QTR
W.D	SAS-FORMAT-WD
WEEKDAY	SAS-FORMAT-WEEKDAY
YEAR	SAS-FORMAT-YEARp
YYQ	SAS-FORMAT-YYQp

[Example of SAS Format to PMML Conversion](#)

For example, the SAS variable purchase_date here is categorized into day of the week values using DOWNNAME format.

```
<DerivedField name="purchase_date+" optype="categorical">
  <!-- SAS Format: DOWNNAME9.0 -->
  <Apply function="SAS-EM-String-Normalize">
    <Constant>9</Constant>
    <Apply function="SAS-FORMAT-DOWNNAMEw">
      <Constant>9</Constant>
      <FieldRef field="purchase_date"/>
    </Apply>
  </Apply>
</DerivedField>
```

[SAS User-Defined Formats in PMML](#)

SAS User-Defined formats are also handled as a function application in PMML. However, the function is explicitly defined in the Transformation Dictionary section, using SAS PROC FORMAT to define the format. Most value, character, and numeric range mapping formats are supported. User-Defined formats involving picture formats, string ranges, multi-labels and fuzz features are not supported.

[Text Normalization in PMML](#)

Enterprise Miner normalizes all category values. This normalization includes trimming leading blanks, truncating a value to a fixed length, and then making the string all uppercase characters. Models are built with normalized values and reference normalized values during scoring. For example, a value like "Yes" is normalized into ?YES?. A PMML scoring engine needs to replicate the normalization process while scoring SAS models. PMML accomplishes string normalization using the SAS-EM-String-Normalize function with a constant parameter for truncation length. An example is shown below.

```
<Apply function="SAS-EM-String-Normalize">
  <Constant>32</Constant>
  <Apply function="SAS-FORMAT-$CHARw">
    <Constant>32</Constant>
    <FieldRef field="URLhit"/>
  </Apply>
```

[Decision Processing in PMML](#)

Decision processing is currently not supported in PMML. Prior probabilities for target levels (priors) specified by a user during training are currently ignored as well, and no adjustments are made to PMML scores.

Scoring A PMML Document

The process of scoring a PMML document is specific to the database vendor and the scoring engine provider. Several vendors, including IBM and Teradata, have developed modules to score PMML models produced by SAS Enterprise Miner. The following example is taken from IBM DB2 Intelligent Miner Scoring, Version 8.1.

The scoring process is represented in two steps:

1. Register a PMML model in DB2. This is done with an SQL statement such as the following:

```
insert into IDMMX.ClassifModels values
  ( 'myRegModel102', IDMMX.DM_impClasFile('\myMachine\myPMMLlib\dmreg.xml') );
```

2. Score a DB2 table with a previously registered model. This is done with statements such as the following:

```
WITH classifView( species, sepallen, sepalwid, petallen, petalwid, classifResult ) AS
(
  SELECT B."species", B."sepallen", B."sepalwid", B."petallen", B."petalwid",
    IDMMX.DM_applyClasModel( C.MODEL, IDMMX.DM_impApplData
      ( rec2xml( 1.0, 'COLATTVAL', '',
        B."sepallen", B."sepalwid", B."petallen", B."petalwid" ) ) )
  FROM "dmairis" B, IDMMX.ClassifModels C
  WHERE C.MODELNAME='myRegModel102'
)
SELECT sepallen, sepalwid, petallen, petalwid, species,
  IDMMX.DM_getPredClass( classifResult ) as pred_species
FROM classifView ;
```

Please refer to the scoring engine vendor documentation for more specific details.