THE POWER TO KNOW.

# SAS® Model Manager 3.1
## Tutorials

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2011. *SAS® Model Manager 3.1: Tutorials*. Cary, NC: SAS Institute Inc.

**SAS® Model Manager 3.1: Tutorials**



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

1st electronic book, July 2011
2nd electronic book, April 2012

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.

# Contents

# About These Tutorials

## Audience

SAS Model Manager is designed for the following users:

- Those who are responsible for developing analytical models.
- Those who are responsible for modeling project management.
- Those who are responsible for model validation and performance testing.
- Scoring officers.
- Analysts.
- SAS administrators and SAS Model Manager Administrators.

## Conventions Used in This Document

The following typographical conventions are used for all text in this document except for syntax:

**bold**
identifies an item in the SAS Model Manager window, a menu item, or a computer pathname.

**`bold monospace`**
identifies text that you enter in a SAS Model Manager window.

*italics*
identifies a book title or a value that is supplied by the user.

**`monospace`**
identifies SAS code.

*Chapter 1*

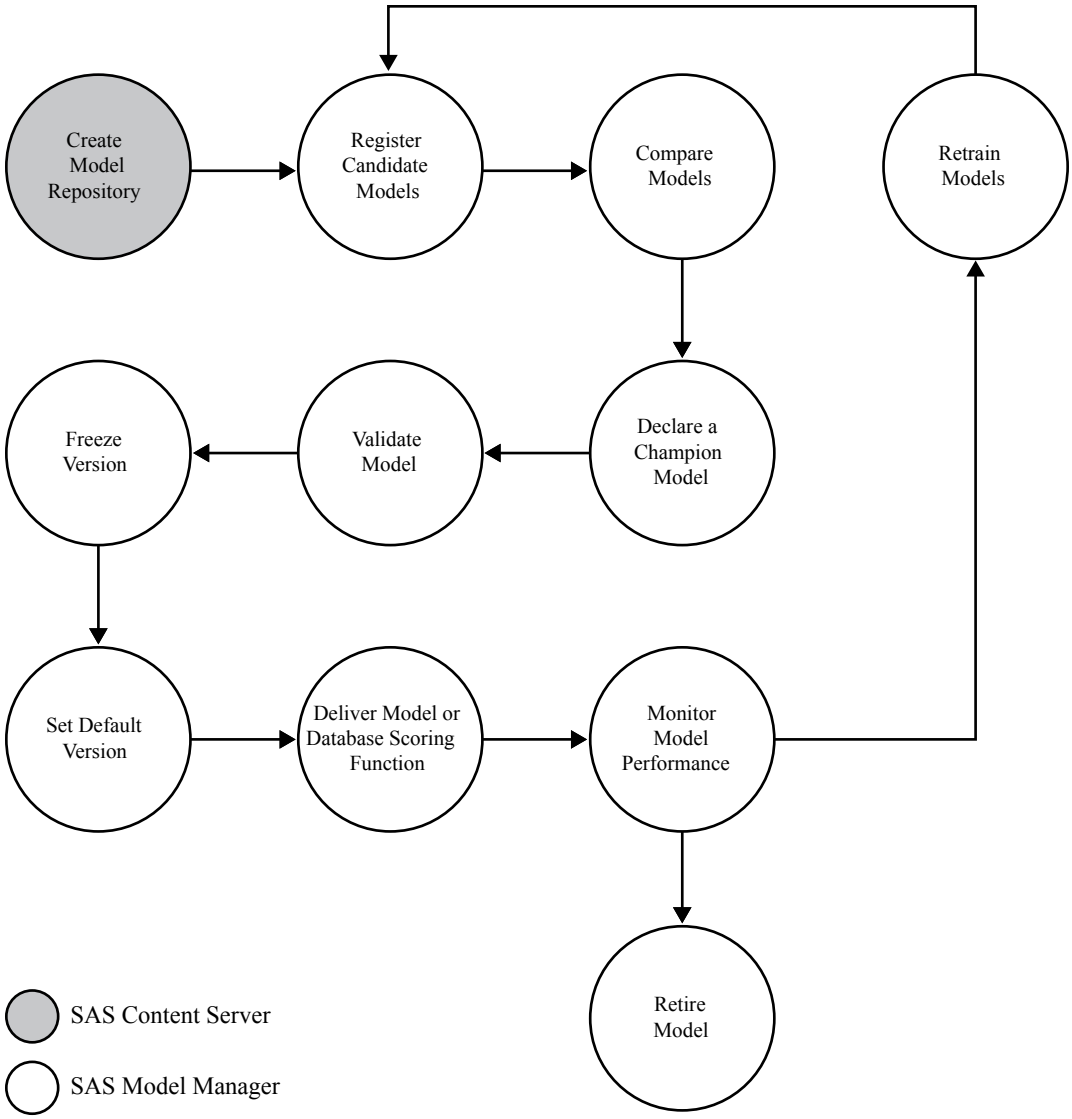
# SAS Model Manager Tutorials



## About SAS Model Manager

SAS Model Manager is a flexible model repository and model management environment for predictive and analytical models. A centralized repository and procedural templates make it easy to manage models and metadata across organizational areas and throughout a model's life cycle. Accountability metrics and validation of analytical steps, from creation to deployment in real time or batch scoring systems, continue until a model is retired. Storing the models in a secure, centralized repository enables you to easily manage life cycle milestones, such as development, test, production, and retirement. The following figure illustrates the model management process that you use in SAS Model Manager:

***Figure 1.1*** *An Example of the Model Management Process*




The goal of a modeling project is to identify a champion model that a scoring application uses to predict an outcome. SAS Model Manager provides tools to evaluate candidate models, declare champion models, and inform your scoring officer that a predictive model is ready for validation or production. You can perform scoring tests for champion and challenger model assessment as well as publish and share the model life cycle and performance data over established reporting channels. You can also run comparative performance benchmarks for the models in your production environment.

## About SAS Model Manager Tutorials

The tutorials for SAS Model Manager cover basic and advanced tasks that are related to model management within an enterprise computing environment. Tutorial folders are created by extracting files from the tutorial ZIP file. You use these data files to become familiar with the following basic tasks that are involved in model management:

- define user groups for assigning and approving life cycle tasks

- define and create the components of the model hierarchy
- import models
- run model reports
- monitor event logging
- register models and select a champion model
- update life cycle milestones or workflow process activities
- run model scoring code in SAS Model Manager
- retrain models

# Install and Register the Tutorial Files

## *About Installing and Registering the Tutorial Files*

The tutorial data sets and models must be installed and registered using SAS Management Console by an administrator who has Write access to the SAS Application Server. A valid SASApp user ID and password are required to install and register the tutorial files.

Some tutorials require files other than data sets and models, such as score code and templates. These files do not need to be registered in the SAS Metadata Repository. The drive where you extract the tutorial ZIP file must be accessible to the SAS Metadata Repository and to tutorial users. Tutorial users can also extract tutorial ZIP files to their local computers in order to access the other files.

## *Download the Tutorial Files*

The ZIP file SMM31Tutorial.zip contains the tutorials' data sets, models, and score code, and is available from **http://support.sas.com/documentation/onlinedoc/modelmgr/**. Before you begin any of the tutorials, extract the tutorial files to a computer that is accessible to the SAS Metadata Server and to the SAS Model Manager users. Follow the steps for using WinZip to extract the files. If you are using a different extraction program, follow that program's instructions for extracting the files.

1. Create a folder on your local computer to store the tutorial files. The instructions refer to this folder as **<drive>**.
2. From **http://support.sas.com/documentation/onlinedoc/modelmgr/**, save SMM31Tutorial.zip to **<drive>**.
3. Open Windows Explorer to **<drive>**. Right-click **SMM31Tutorial.zip** and select **Open with WinZip**.
4. From the WinZip window, click the **Extract** button. The Extract dialog box appears.

   *Note:* If you are running Windows 7, click the arrow on the **Unzip** button to open the Unzip from WinZip File Folder window.
5. In the **Extract to** box, select **<drive>** and click **Extract**.

   *Note:* If you are running Windows 7, select **<drive>** from the Unzip from WinZip File Folder window.

You can find the files for each tutorial in the respective tutorial folder (for example, **<drive>**\Tutorial2 or **<drive>**\Tutorial3).

## Prepare Tutorial 2 Data Sets and Models

### The Required Tutorial Files

Tutorial 2 requires the following files and folders in the ***<drive>*\Tutorial2\Samples** folder:

- delinquency_project_input.sas7bdat
- delinquency_project_output.sas7bdat
- delinquency_scoring_input.sas7bdat
- delinquency_scoring_output.sas7bdat
- delinquency_test.sas7bdat
- delinquency_train.sas7bdat
- The **model1** folder contains these model files:
  - modelin1.sas.7bdat
  - modelout1.sas.7bdat
  - om.sas.7bdat
  - result1.sas.7bdat
  - score1.sas.7bdat
  - target1.sas.7bdat
- The **model2** folder contains these model files:
  - modelin2.sas.7bdat
  - modelout2.sas.7bdat
  - ot.sas.7bdat
  - result2.sas.7bdat
  - score2.sas.7bdat
  - target2.sas.7bdat
- The **model3** folder contains these model files:
  - modelin3.sas.7bdat
  - modelout3.sas.7bdat
  - result3.sas.7bdat
  - score3.sas.7bdat
  - target3.sas.7bdat

### Register the Tutorial 2 Files in SAS Management Console

Use SAS Management Console to define a data library and register the tables. Follow these steps:

1. Start and log on to SAS Management Console as a SAS Administrator with the role **Metadata Server: Operation or Metadata Server: Unrestricted**.
2. Open the New Library Wizard to a define the data library. Expand **Environment Management** ⇨ **Data Library Manager** on the **Plug-ins** tab. Right-click **Libraries** and select **New Library** from the pop-up menu.
3. In the New Library Wizard, create a SAS library.
   a. Navigate to **Resource Templates** ⇨ **SAS Data**

      Select **SAS BASE Library** and click **Next**.

   b. Specify **`MM Tutorial-2`** in the **Name** field and click **Browse**. In the Select a Location dialog box, double-click **Model Manager** and then click the **New folder** icon. Create the folder **`Tutorial2`** and make **Tutorial2** the active folder. Click **OK**. Click **Next**.

   c. Select **SASApp** from the **Available servers** box and click  to move the server name to the **Selected servers** list. Click **Next**.

   d. Specify **`smm3tor2`** for the libref and click **New**.

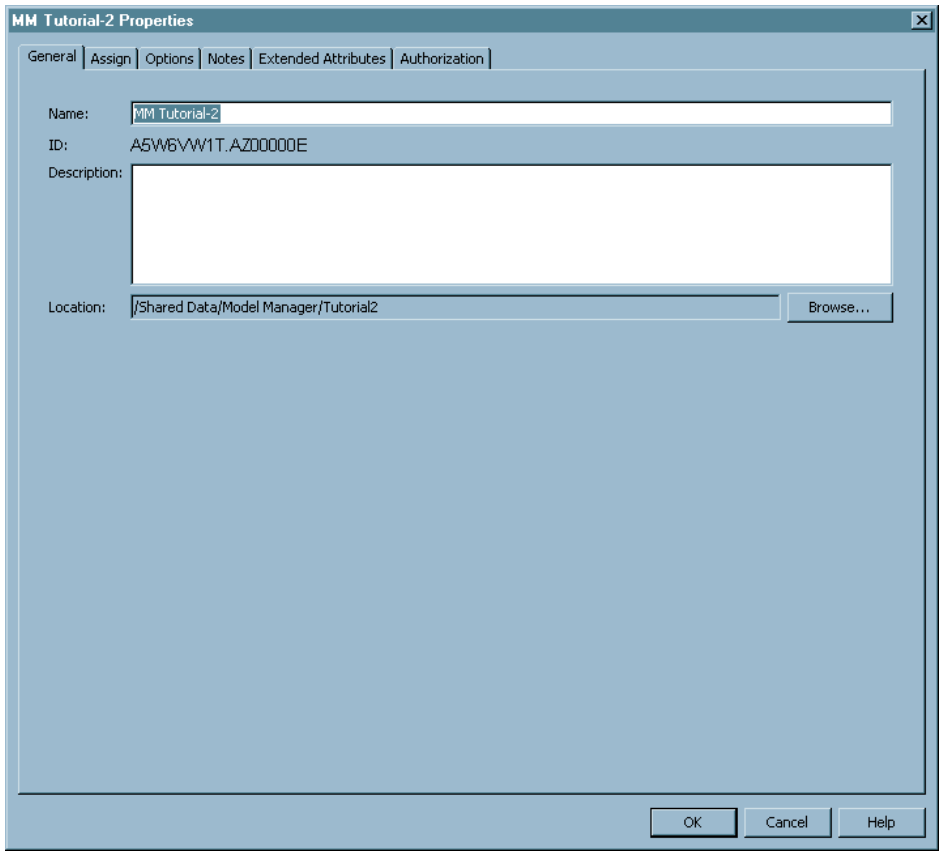
   e. Select the path for the Tutorial 2 files in the **Available items** box and click  to move the path to the **Selected items** list. Click **Next**.

   f. Click **Finish**. Verify that the library MM Tutorial-2 is a library in the **Libraries** folder.
4. Register the data tables in the metadata repository.
   a. Right-click **MM Tutorial-2** under the **Libraries** folder, select **Register Tables** from the pop-up menu, and click **Next**.

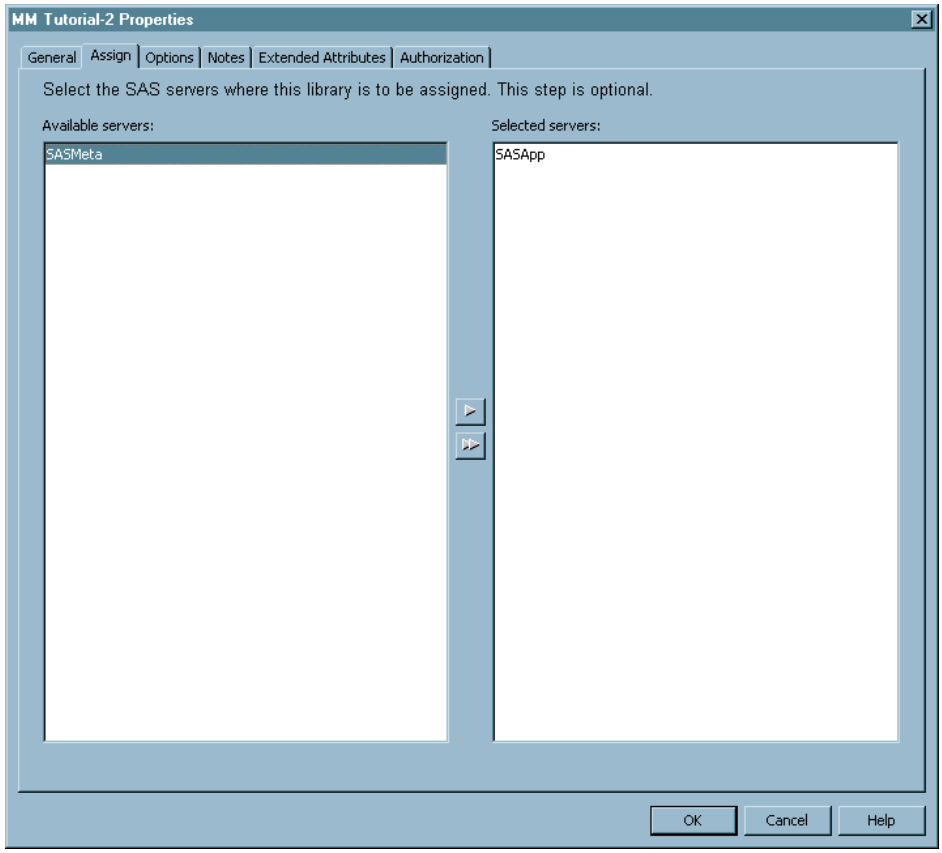
   b. If prompted, specify a user ID and password that has access to the metadata server and click **OK**.

   c. Click **Select All Tables**, click **Next**, and click **Finish**.
5. Verify that table metadata was created and close SAS Management Console. Right-click **MM Tutorial-2** and select **Properties**.

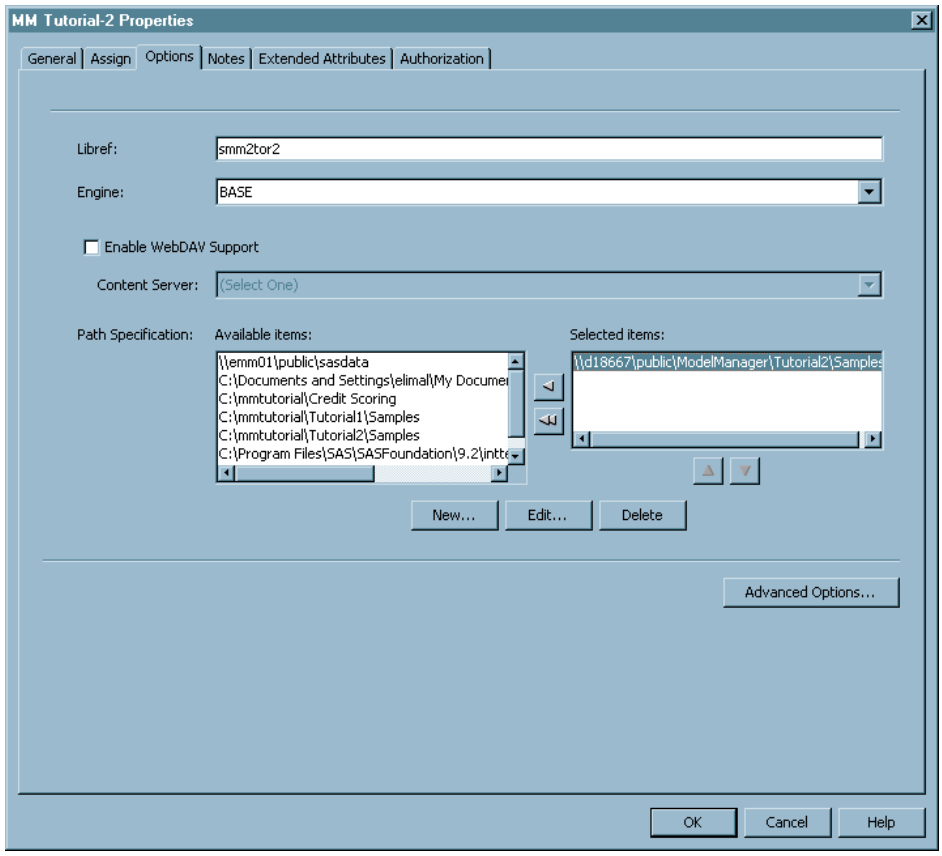
   Verify the tutorial name and location.


MM Tutorial-2 Properties
General
Assign
Options
Notes
Extended Attributes
Authorization
Name:
MM Tutorial-2
ID:
A5W6VW1T.AZ00000E
Description:
Location:
/Shared Data/Model Manager/Tutorial2
Browse...
OK
Cancel
Help

Click the **Assign** tab. Verify that the appropriate server is in the **Selected servers** list.




Click the **Options** tab. Verify the libref, the engine, and the path specification.




## Prepare Tutorial 3 Data Sets and Models

### The Required Tutorial 3 Files

The SAS data sets and models that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

This tutorial requires the following files and folders in the **`<drive>\Tutorial3\Samples`** folder:

- hmeq_project_input.sas7bdat
- hmeq_project_output.sas7bdat
- hmeq_score_input.sas7bdat
- hmeq_test.sas7bdat
- hmeq_train.sas7bdat
- \Neural\Neutral.xml
- \Reg1\miningResult.spk
- \Tree1\miningResult.spk

### *Register the Tutorial 3 Files in SAS Management Console*

Use SAS Management Console to define a data library and register the tables. Follow these steps:

1. Open SAS Management Console and log on to the SAS Metadata Server. Your user ID must be authorized to modify libraries in the metadata server.
2. Open the New Library Wizard to define the data library. Expand **Environment Management** ⇨ **Data Library Manager** on the **Plug-ins** tab. Right-click **Libraries** and select **New Library** from the pop-up menu.
3. In the New Library Wizard, create a SAS library.
   a. Navigate to **Resource Templates** ⇨ **SAS Data**
   b. Select **SAS BASE Library** and click **Next**.
   c. Specify `MM Tutorial-3` in the **Name** field and click **Next**.
   d. (Optional) If more than one server exists, select a server. Click **Next**.
   e. Specify `smm3tor3` for the libref and click **New**.
   f. Specify the server folder that you previously created, `<drive>\Tutorial3\Samples`, for the path specification and click **OK** twice.
   g. Click **Next** and **Finish**.
4. Register the data tables in the metadata repository.
   a. Right-click **MM Tutorial-3** under the **Libraries** node, select **Register Tables** from the pop-up menu, and click **Next**.
   b. If prompted, specify the metadata server, test the server connection, and click **OK**. Click **Next**.
   c. Click **Select All Tables**, click **Next**, and click **Finish**.
5. Verify that table metadata was created and close SAS Management Console. Select **MM Tutorial-3** under the **Libraries** node and examine the right pane.
6. Authorize the **MM Tutorial Assignees** group and the **MM Tutorial Approvers** group to read and write to the library:
   a. Right-click the **MM Tutorial-3** library and select **Properties**.
   b. Click the **Authorization** tab and click the **Add** button.
   c. Select **MM Tutorial Assignees** from the **Available Identities** list and click  to move the group to the **Selected Identities** list.
   d. Select **MM Tutorial Approvers** from the **Available Identities** list and click  to move the group to the **Selected Identities** list.
   e. Click **OK** twice.

## *Prepare Tutorial 6 Data Sets and Models*

### *The Required Tutorial 6 Files*

The SAS data sets and models that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

This tutorial requires the following files and folders in the **<drive>\Tutorial6\Samples** folder:

- hmeq_project_input.sas7bdat
- hmeq_project_output.sas7bdat
- hmeq_score_input.sas7bdat
- hmeq_score_output.sas7bdat
- hmeq_test.sas7bdat
- hmeq_train.sas7bdat
- VarImportance.sas
- \Model6\importance6.sas7bdat
- \Model6\modelinput6.sas7bdat
- \Model6\modeloutput6.sas7bdat
- \Model6\nodestat6.sas7bdat
- \Model6\path6.sas7bdat
- \Model6\rules6.sas7bdat
- \Model6\score6.sas
- \Model6\target6.sas7bdat

### *Register the Tutorial 6 Files in SAS Management Console*

Use SAS Management Console to define a data library and register the tables. Follow these steps:

1. Open SAS Management Console and log on to the SAS Metadata Server. Your user ID must be authorized to modify libraries in the metadata server.
2. Open the New Library Wizard to define the data library. Click **Environment Management** ⇨ **Data Library Manager** on the **Plug-ins** tab. Right-click **Libraries** and select **New Library** from the pop-up menu.
3. In the New Library Wizard, create a SAS library.
   a. Navigate to **Resource Templates** ⇨ **SAS Data**

      Select **SAS BASE Library** and click **Next**.
   b. Specify **MM Tutorial-6** in the **Name** field.
   c. From the **Location** box, click **Browse**. Navigate to the **Model Manager** folder. Click the **New folder** icon and enter **Tutorial6**. Click the dialog box edges, click **Tutorial6**, and click **OK**.
   d. (Optional) If more than one server exists, select a server in the **Available servers** list and click  to move the server name to the **Selected servers** list.
   e. Specify **smm3tor6** for the libref and click **New**.
   f. Specify the server folder that you previously created, ***drive*\Tutorial6\Samples**, for the path specification and click **Next**.
   g. Click **Finish**.
4. Register the data tables in the metadata repository.

a. Right-click **MM Tutorial-6** under the **Libraries** node, select **Register Tables** from the pop-up menu, and click **Next**.

b. If prompted, specify the user ID and password for the server and click **OK**. Then click **Next**.

c. Click **Select All Tables**, click **Next**, and click **Finish**.

5. Verify that table metadata was created and close SAS Management Console. Select **MM Tutorial-6** under the **Libraries** node and examine the right pane.

## *Prepare Tutorial 7 Data Sets and Models*

### *The Required Tutorial 7 Files*

The SAS data sets that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

This tutorial requires the following files and folders in the **`<drive>\Tutorial7\Samples`** folder:

- hmeq_2010Q2.sas7bdat
- hmeq_2010Q3.sas7bdat
- hmeq_2010Q4.sas7bdat
- hmeq_2011Q1.sas7bdat

### *Register the Tutorial 7 Files in SAS Management Console*

Use SAS Management Console to define a data library and register the tables. Follow these steps:

1. Open SAS Management Console and log on to the SAS Metadata Server. Your user ID must be authorized to modify libraries in the metadata server.
2. Open the New Library Wizard to define the data library. Click **Environment Management** ⇨ **Data Library Manager** on the **Plug-ins** tab. Right-click **Libraries** and select **New Library** from the pop-up menu.
3. In the New Library Wizard, create a SAS library.

   a. Navigate to **Resource Templates** ⇨ **SAS Data**

   Select **SAS BASE Library** and click **Next**.

   b. Specify **`MM Tutorial-7`** in the **Name** box.

   c. From the **Location** box, click **Browse**. Navigate to the **Model Manager** folder. Click the **New folder** icon and enter **`Tutorial7`**. Click the dialog box edges, click **Tutorial7**, and click **OK**.

   d. (Optional) If more than one server exists, select a server in the **Available servers** list and click  to move the server name to the **Selected servers** list.

   e. Specify **`smm3tor7`** for the libref and click **New**.

   f. Specify the server folder that you previously created, **`<drive>\Tutorial7\Samples`**, and click  to move the path to the **Selected items** box.

   g. Click **Next** and **Finish**.

4. Register the data tables in the metadata repository.
   a. Right-click **MM Tutorial-7** under the **Libraries** node, select **Register Tables** from the pop-up menu, and click **Next**.
   b. If prompted, specify the user ID and password to the metadata server and click **OK**. Then click **Next**.
   c. Click **Select All Tables** and click **Next**.
   d. Click **Finish**.
5. Verify that table metadata was created. Select **MM Tutorial-7** under the **Libraries** node and examine the right pane.

## Prepare Tutorial 8 Data Sets and Models

### The Required Tutorial 8 Files

The SAS data sets and models that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

This tutorial requires the following files and folders in the **`<drive>\Tutorial8\Samples`** folder:

- score_input.sas7bdat
- score_output.sas7bdat

### Register the Tutorial 8 Files in SAS Management Console

Use SAS Management Console to define a data library and register the tables. Follow these steps:

1. Open SAS Management Console and log on to the SAS Metadata Server. Your user ID must be authorized to modify libraries in the metadata server.
2. Open the New Library Wizard to define the data library. Click **Environment Management** ⇨ **Data Library Manager** on the **Plug-ins** tab. Right-click **Libraries** and select **New Library** from the pop-up menu.
3. In the New Library Wizard, create a SAS library.
   a. Navigate to **Resource Templates** ⇨ **SAS Data**

      Select **SAS BASE Library** and click **Next**.
   b. Specify **`MM Tutorial-8`** in the **Name** box.
   c. From the **Location** box, click **Browse**. Navigate to the **Model Manager** folder. Click the **New folder** icon and enter **`Tutorial8`**. Double-click **Tutorial8**. Click **OK**.
   d. (Optional) If more than one server exists, select a server in the **Available servers** list and click  to move the server name to the **Selected servers** list.
   e. Specify **`smm3tor8`** for the libref and click **New**.
   f. Specify the server folder that you previously created, **`<drive>\Tutorial8\Samples`**, for the path specification and click  to move the path to the **Selected items** box.
   g. Click **Next** and **Finish**.

4. Register the data tables in the metadata repository.
   a. Right-click **MM Tutorial-8** under the **Libraries** node, select **Register Tables** from the pop-up menu, and click **Next**.
   b. If prompted, specify the user ID and password to the metadata server and click **OK**. Then click **Next**.
   c. Click **Select All Tables** and click **Next**.
   d. Click **Finish**.
5. Verify that table metadata was created. Select **MM Tutorial-8** under the **Libraries** node and examine the right pane.

# Prepare for Using SAS Workflow

## *Overview*

SAS Workflow provides services that work together to model, automate, integrate, and streamline business processes. It provides a platform for more efficient and productive business solutions.

SAS Workflow is used by SAS solutions that benefit from business process management. SAS Workflow Studio is a desktop client application that is used to design and deploy workflow process definitions. The SAS middle tier hosts the workflow engine and the workflow services as part of the SAS Web Infrastructure Platform. The SAS Model Manager Workflow Console is used to manage the workflow processes that are associated with modeling projects and versions.

From the SAS Model Manager client application, you can view workflow instances, create a new workflow instance for a project or version, and view your workflow inbox to work with activities, depending on the user permissions. The option that is selected and the user permissions determine the category view and content that are displayed when Workflow Console is launched. SAS Model Manager administrators can view the Process Definitions, Instances, and Activities category views of Workflow Console. SAS Model Manager users and advanced users can view only the Activities category view. For more information about user permissions, see "Configuring Users, Groups, and Roles" in Chapter 3 of *SAS Model Manager: Administrator's Guide*.

To use SAS Workflow with SAS Model Manager tutorials, the following prerequisites must be met:

1. SAS Workflow Engine, SAS Workflow Services, and SAS Workflow Studio must be installed and configured. For more information, see *SAS Intelligence Platform: Installation and Configuration Guide*.
2. If you want to receive the e-mail notifications for a tutorial workflow process, you must configure alert notifications using SAS Management Console. For more information, see "Configure Alert Notifications" on page 14.
3. Workflow process definitions must be made available using SAS Workflow Studio. For more information, see "Make the Workflow Process Definitions Available" on page 15.

## Configure Alert Notifications

### About Alert Notifications

To enable workflow participants to receive alert notifications from SAS Workflow when performing the tutorials, you must configure the **`E-mail`** notification type in SAS Management Console. After you have configured the alert notifications, you can then use the Notify Participant policy and other workflow notification policies for workflow process activities in SAS Workflow Studio. The notifications setting in SAS Management Console is a global setting. Preferences and notifications can also be configured for individual users.

The Send Notification By Data Object policy in SAS Workflow Studio integrates with the SAS Web Infrastructure Platform's Notification Service. Recipients are notified according to their preferences (e-mail or portlets).

### Global Alert Notifications

To enable the e-mail notification type for all users, follow these steps:

1. Log on to SAS Management Console as a SAS administrator.
2. On the **Plug-ins** tab, navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure**.
3. Right-click **SAS Application Infrastructure** and select **Properties**.
4. Click the **Settings** tab.
5. Select **Notifications** in the left panel. Use the menus or text fields to set the property.
6. Select the **`E-mail`** notification type.
7. Click **OK**.
8. To apply this setting and make it available, restart the SAS Web Infrastructure Platform Services, and applications that use the changed property, such as SAS Model Manager Workflow Console and SAS Workflow.

### Individual User Alert Notifications

You can use SAS Preferences Manager to override the default notification delivery type for your user account. SAS Preferences Manager is a Web application that provides a central facility for users to manage their preferences and settings. The default notification type after the deployment of SAS 9.3 is the alerts portlet.

To modify your notification delivery preference, follow these steps:

*Note:* For the SAS Model Manager tutorials, you need only the e-mail notification delivery type.

1. Enter the URL **`http://`*`host-name`*`:`*`port`*`/SASPreferences`** in your browser window to launch the SAS Preferences Manager. Replace the values for host-name and port based on the location of the configured SAS Web Infrastructure Platform.
2. Enter the user ID and password for the user account that you use to access SAS Web applications and SAS Model Manager.
3. Select **General** ⇨ **Notifications**.
4. Select a format type for the E-mail notifications. The options are **`HTML-formatted e-mail`** and **`Plain-text e-mail`**.

5. Select the notification types from the **Available** list and click to add the selected notification types. The available options are the following:
   - Via e-mail
   - My alerts portlet
   - Via SMS text message
   - Via digested e-mail

   **TIP** To remove a notification type, select the type from the list and click to remove the selected item.
6. Click **Apply** to update the notification settings and click **OK** to save the changes.

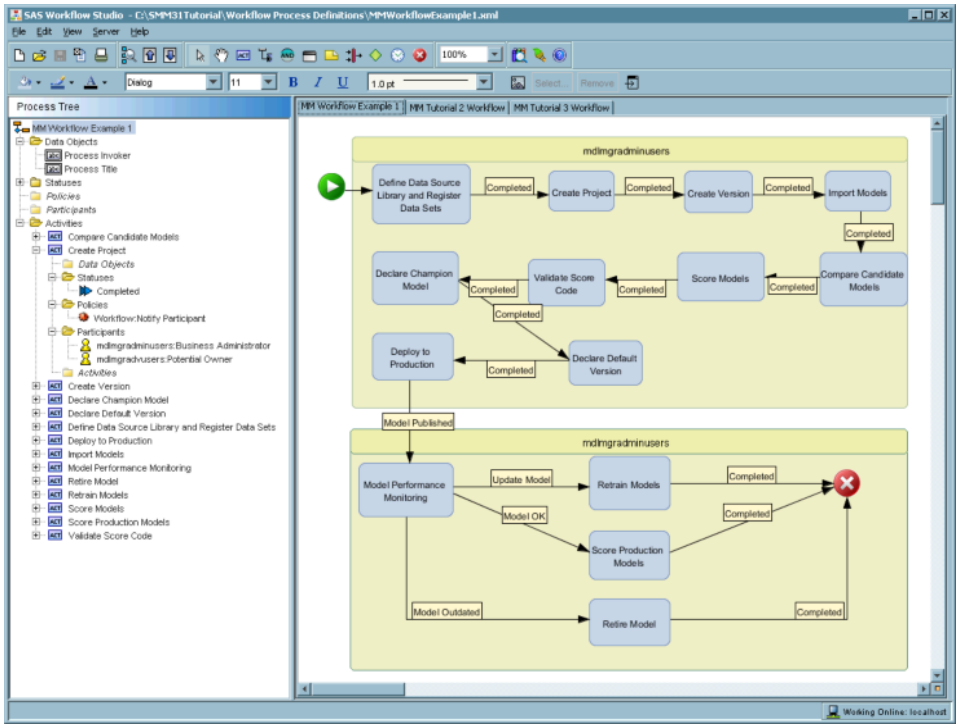
## *Make the Workflow Process Definitions Available*

### *Overview*

To use SAS Workflow with tutorials, you must make the process definitions available to SAS Model Manager. After the process definitions are made available, the SAS Model Manager administrator can use Workflow Console to create workflow instances to be used with SAS Model Manager.

To save the tutorial workflow process definitions to the Workflow repository, follow these steps:

1. From SAS Workflow Studio, select **File** ⇨ **Open** and navigate to the location where you extracted the tutorial files. For example, **`C:\SMM31Tutorial\`**. Open the subfolder **`Workflow Process Definitions`** and select the file. For example, **`MMWorkflowExample1.xml`**.




2. Log on to the server as a SAS administrator or SAS Model Manager administrator.
3. Add the tag attribute of **`mmapi`** to the process definition file properties.

4. Upload the process definition.
5. Repeat steps 1 through 4 for each workflow process definition. There are workflow process definitions for tutorial 2 and tutorial 3.
6. Verify that the process definitions appear in the Process Definitions category view of SAS Model Manager Workflow Console.

For more information, see "Deploying and Maintaining Processes" in the *SAS Workflow Studio 1.1: User's Guide*. The user's guide is available from the URL **http://support.sas.com/documentation/onlinedoc/workflow/**.

### Log On to the Server

With SAS Workflow Studio, you can manage only locally stored workflow process definitions on your system until you have logged on to the SAS Content Server. After you are connected, you can access additional process definitions that are stored in the SAS Content Server.

To log on to the server, follow these steps:

1. From the **Server** menu, select **Logon**.
2. In the Log On window, select the host-name from the **SAS environment** drop-down list.

   *Note:* For more information, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*.
3. Enter a user ID and password, and click **Log On**.

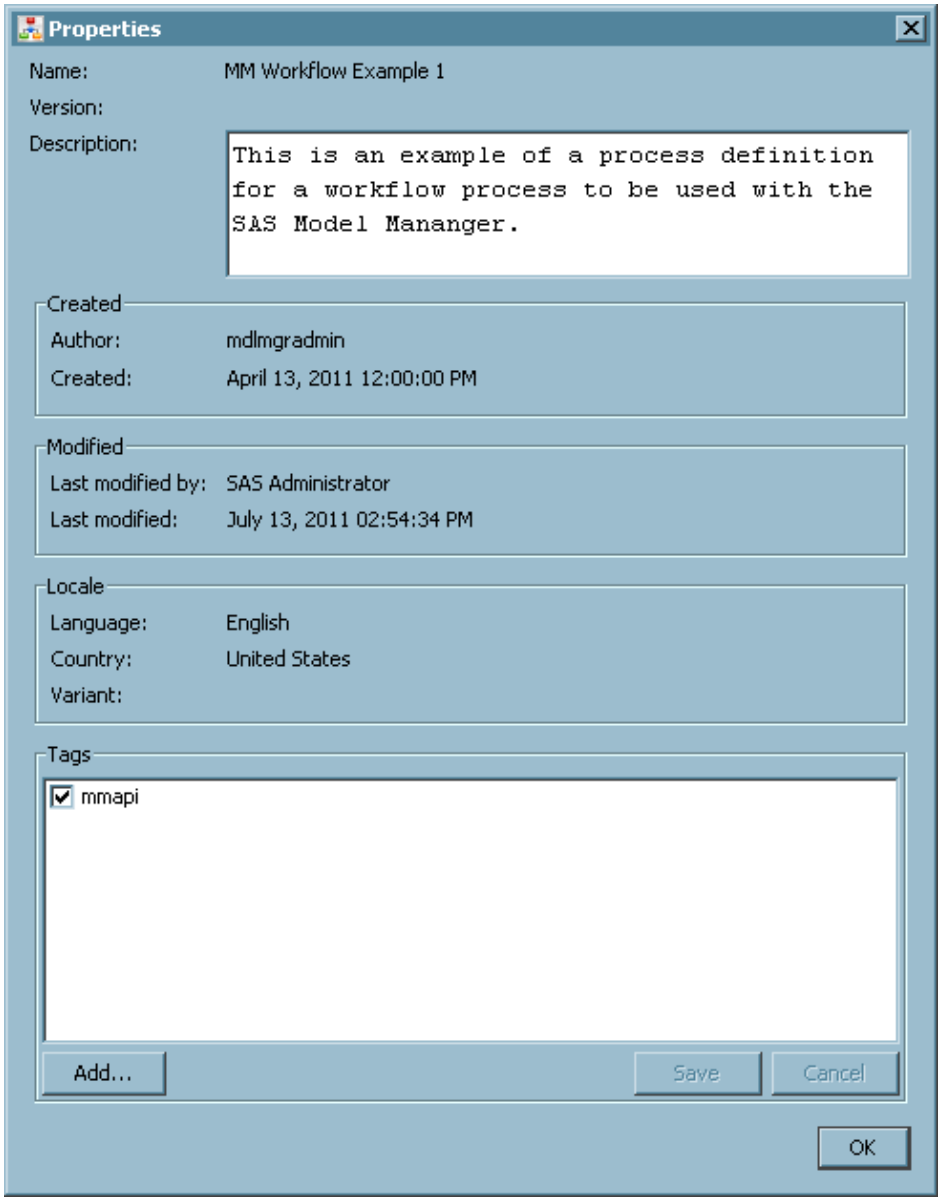
### Add Tag Attributes to a Process Definition

Only the process definitions in the Workflow repository that have the **mmapi** tag attribute that is specified in the file properties are available to SAS Model Manager in the Workflow Console. The Workflow repository is located on the SAS Content Server.

To add a tag attribute to the file properties of a process definition template in SAS Workflow Studio, follow these steps:

1. Select **File** ⇨ **Properties** and click **Add**.
2. Enter the tag value of **mmapi**.

   *Note:* The file properties are case sensitive. This value must be lowercase.

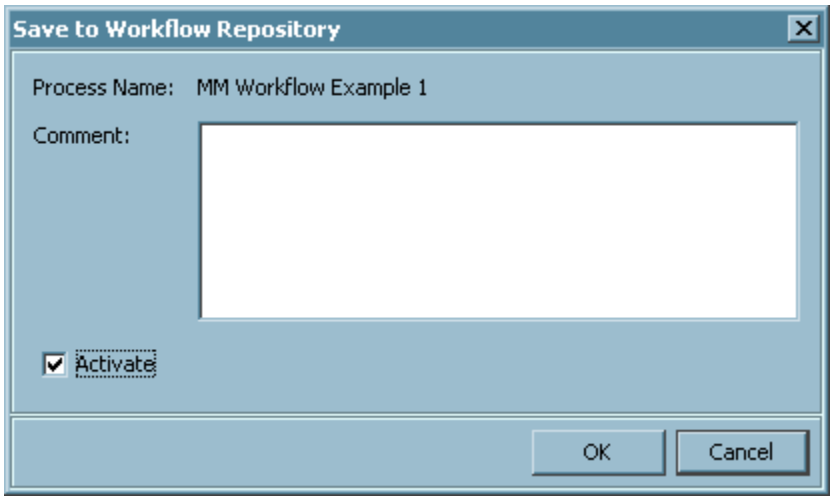



3. Click **OK** twice.

### *Upload a Process Definition*

To upload a process, follow these steps:

1. From the **Server** menu, select the **Save to Repository** menu option. The Save to Workflow Repository window appears.
2. (Optional) Enter relevant comments to associate with the process definition.
3. Select the **Activate** option if you want to make the current version of the workflow process definition available for use in the Workflow repository by applications, such as SAS Model Manager Workflow Console.

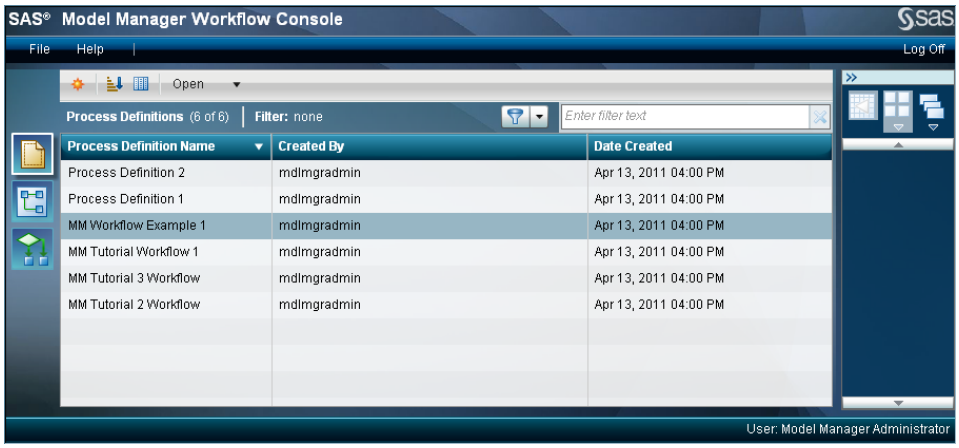



4. Click **OK**.

### *Verify That the Process Definitions Are Available in the Workflow Console*

To verify that the workflow process definitions are available in the Workflow Console, follow these steps:

1. Enter the URL **`http://`*`host-name`*`:`*`port`*`/SASModelManagerWorkflow`** in your Web browser.
2. Enter the user ID and password for a SAS Model Manager administrator.
3. Verify that the uploaded process definition appears in the Process Definitions category view.



*Chapter 2*

# Tutorial 1: Create a Life Cycle Template and a Workflow Process Instance



## Overview of Life Cycle Templates and Roles

A SAS Model Manager project consists of one or more versions. A version is a time-based container for SAS Model Manager projects. For example, you might have versions for both 2010 and 2011. The 2010 version contains the champion home equity model and the 2011 version contains the resources for developing a new home equity model.

Each version has a life cycle that is associated with it to track the progress of selecting a champion model and monitoring the model's performance. The life cycle contains milestones such as development, test, and production. Associated with each milestone are tasks that you perform to complete a milestone. When you create a version, you select a life cycle template that you want to use for the version life cycle. The life cycle template for your version must be available to SAS Model Manager before you create a version.

You use the SAS Model Manager Template Editor to create a life cycle template with milestones and tasks that are specific to your modeling project. You can create a new life cycle template or start with a sample life cycle template and modify the template. SAS Model Manager provides several sample life cycle templates. After the template is complete, you can use the template in SAS Model Manager by uploading the template to

the SAS Content Server. You can save a backup copy of the template to a local or network location.

In SAS Model Manager, you can view life cycle templates from the Life Cycle perspective. Any user-defined template in the Life Cycle perspective can be used as a life cycle when you create a version.

Any users or groups who need to update the version life cycle status must be assigned the appropriate life cycle roles using the SAS Management Console User Manager plug-in:

- Model Manager: Life Cycle Participant Usage (participant)
- Model Manager: Life Cycle Assignee Usage (assignee)
- Model Manager: Life Cycle Approval Usage (approver)

A best practice is to assign these roles only to groups and not to users. Assigning roles to groups provides flexibility when you need to add or remove users who are responsible for life cycle tasks. However, because you can assign these roles to users as well as groups, this tutorial states that both users and groups can be assigned to tasks.

When you open the SAS Model Manager Template Editor, users or groups that are assigned to the participant role appear in the **Participants** list. Only those users and groups in the **Participants** list can be considered to be assignees or approvers. When the template is selected as the life cycle for a version, only those users or groups can update the milestone and task properties.

This tutorial creates a Model Manager Tutorial Users group by using SAS Management Console. After you create the Model Manager Tutorial Users group, you create a life cycle template that can be used for the SAS Model Manager tutorials.

# Create Groups for Use with the SAS Model Manager Tutorial

## *Create a SAS Model Manager Assignee Group*

In this exercise, a SAS administrator creates a group in SAS Management Console for SAS Model Manager assignees. Any member of this group is able to update the status of a life cycle task if that group is specified as a value for the task **Assignee** property.

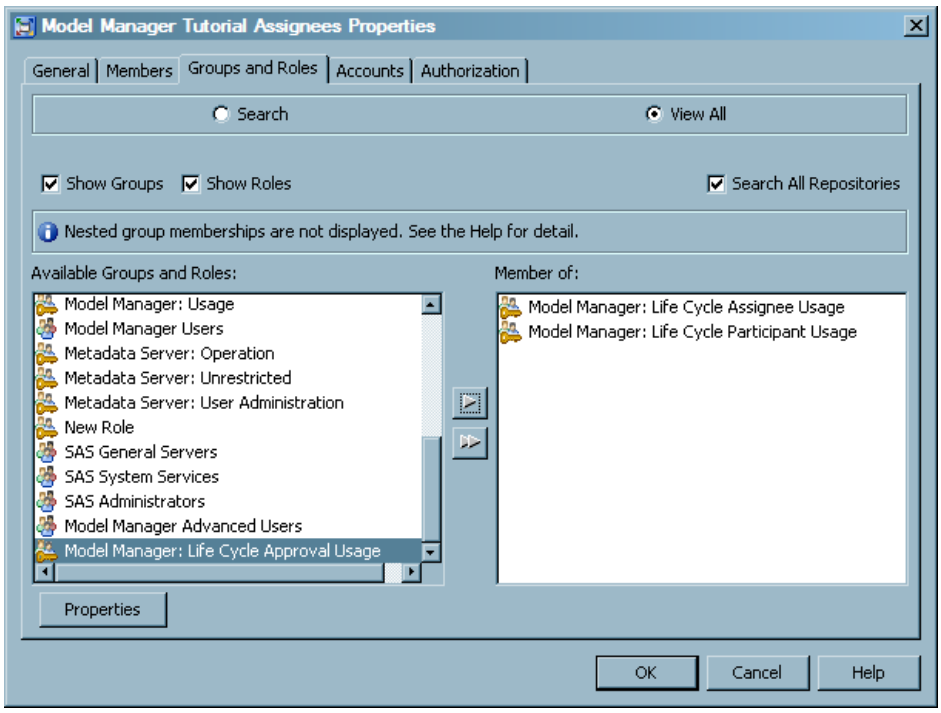
1. Start and log on to SAS Management Console as a SAS Administrator whose role enables you to update the metadata server user administration.
2. On the **Plug-ins** tab, right-click **User Manager** and select **New** ⇨ **Group**. The New Group Properties window appears with the **General** tab.
3. In the **Name** field, type **`MM Tutorial Assignees`**.
4. In the **Display Name** field, type **`MM Tutorial Assignees`**.
5. In the **Description** field, type **`A group for SAS Model Manager users who can be assigned to complete tasks.`**
6. Click the **Members** tab.
7. From the **Available Identities** list, select **Model Manager Advanced Users**, **Model Manager Administrator Users**, and **Model Manager Users** to add to this group.

For each user who needs to be assigned to the group, select the user and click ▶ to move the user to the **Current Members** list.

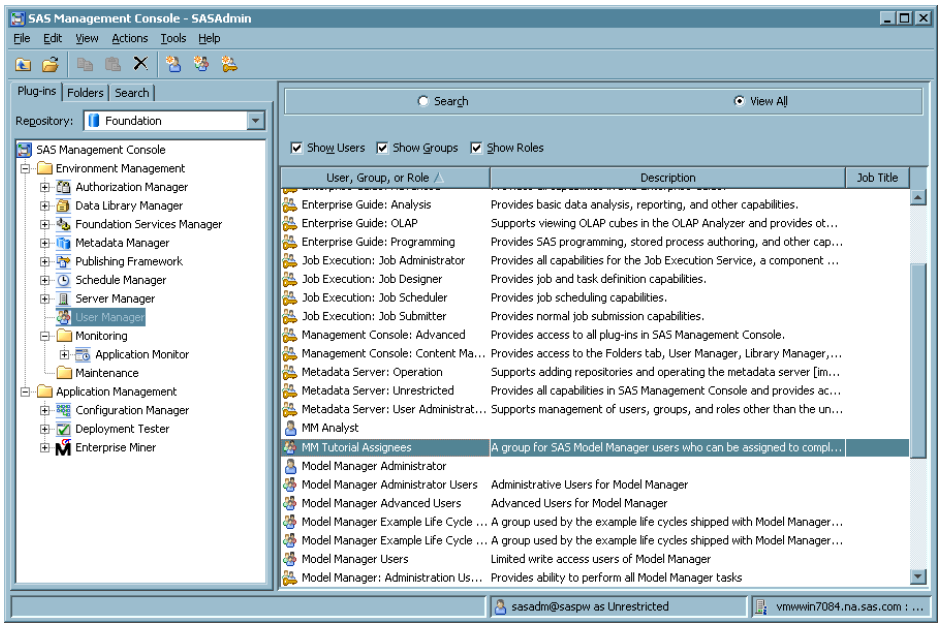
8. Click the **Groups and Roles** tab. Ensure that the **Show Roles** box is checked. Select the following roles and click ▶ to move the roles to the **Member of** list:

   - **Model Manager: Life Cycle Participant Usage**
   - **Model Manager: Life Cycle Assignee Usage**

   Here is an example of the **Groups and Roles** tab.




9. Click **OK**. Here is an example of the group **MM Tutorial Assignees** listed as a group in SAS Management Console.

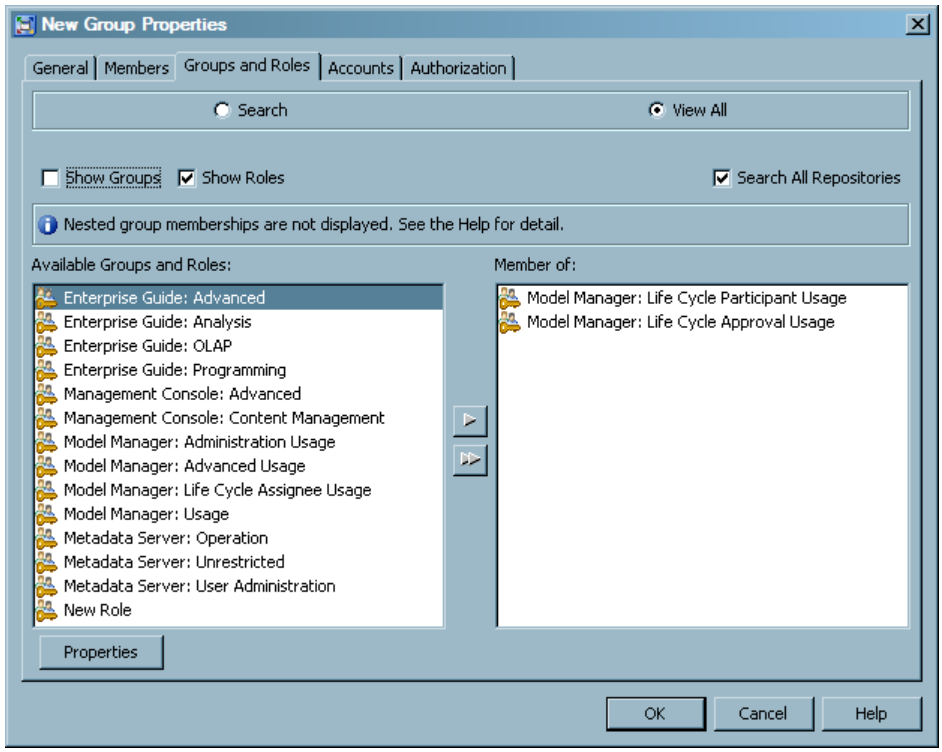


### Create a SAS Model Manager Approver Group

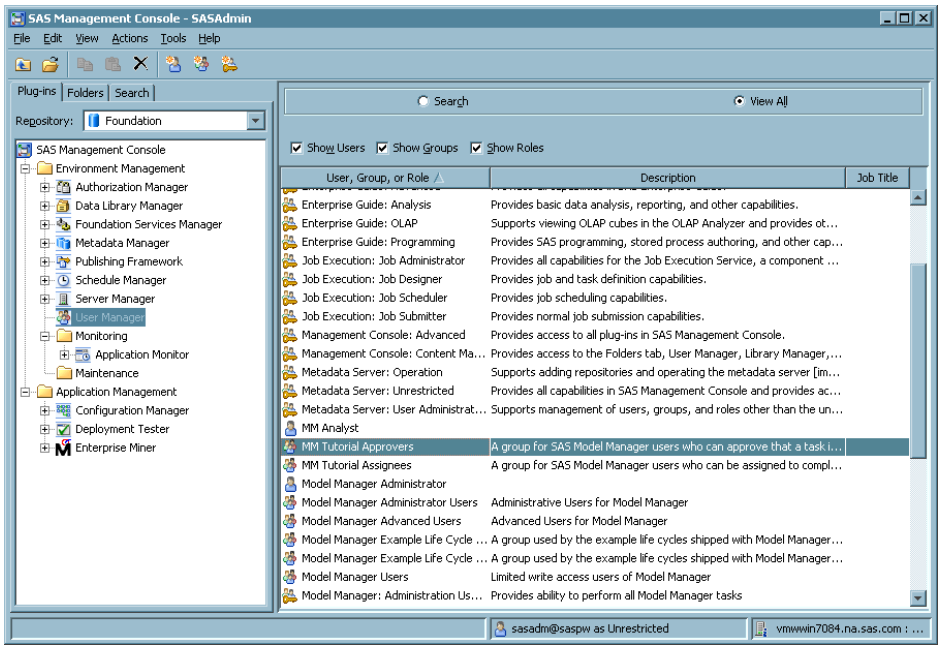
In this exercise, a SAS administrator creates a group in SAS Management Console for SAS Model Manager approvers. Any member of this group is able to update the **Approved** status of a life cycle task if that group is specified as a value for the task **Approver** property.

1. Start and log on to SAS Management Console as a SAS Administrator whose role enables you to update the metadata server user administration.
2. On the **Plug-ins** tab, right-click **User Manager** and select **New** ⇨ **Group**. The New Group Properties window appears with the **General** tab.
3. In the **Name** field, type **`MM Tutorial Approvers`**.
4. In the **Display Name** field, type **`MM Tutorial Approvers`**.
5. In the **Description** field, type **`A group for SAS Model Manager users who can approve that a task is complete.`**
6. Click the **Members** tab.
7. From the **Available Identities** list, select **Model Manager Advanced Users** and **Model Manager Administrator** to add to this group. For each user who needs to be assigned to the group, select the user and click ▷ to move the user to the **Current Members** list.
8. Click the **Groups and Roles** tab. Ensure that the **Show Roles** box is checked. Select the following roles and click ▷ to move the roles to the **Member of** list:
   - **Model Manager: Life Cycle Participant Usage**
   - **Model Manager: Life Cycle Approval Usage**

   Here is an example of the **Groups and Roles** tab.



9. Click **OK**. Here is an example of the group **MM Tutorial Approvers** listed as a group in SAS Management Console.




# Create a Life Cycle Template

In this exercise, you use the SAS Model Manager Template Editor to create a user-defined life cycle template from a sample template. SAS Model Manager provides sample templates that you can use to start your user-defined template. This tutorial uses the Basic.xml template.

## *Start SAS Model Manager*

To run SAS Model Manager, follow these steps:

1. On your client machine, start the SAS Model Manager client.
2. Log on to SAS Model Manager as a member of the **Model Manager Advanced Users** group or **Model Manager Administrator Users** group.


### *Create a New Life Cycle Template*

This task uses the Basic.xml sample template that is provided by SAS Model Manager and modifies it to create a new life cycle template.

1. Open the SAS Model Manager Template Editor (Template Editor). Select **Tools** ⇨ **Manage Templates** in the SAS Model Manager window.
2. In the Template Editor, open the Basic.xml sample life cycle template. Select **File** ⇨ **Browse** ⇨ **Browse Templates** ⇨ **Basic.xml** and click **Open**.
3. Modify the template properties. Specify the following properties:

   **Name**
   Replace the name with **`Tutorial Life Cycle`**.

   **Description**
   Replace the description with **`A life cycle for the tutorial.`**
4. Save the template to your local computer by selecting **File** ⇨ **Save As**. In the Save dialog box, select the location on your local computer. Enter **`TutorialLifeCycle.xml`** as the filename and click **Save**. Click **OK** in the information dialog box to save the template as an XML file.
5. Using a text editor, open the life cycle template XML file that you just saved. If the version attribute on the <LifeCycleTemplate> does not have a value of "1", replace the value with the value "1". Here is the <LifeCycleTemplate> element:

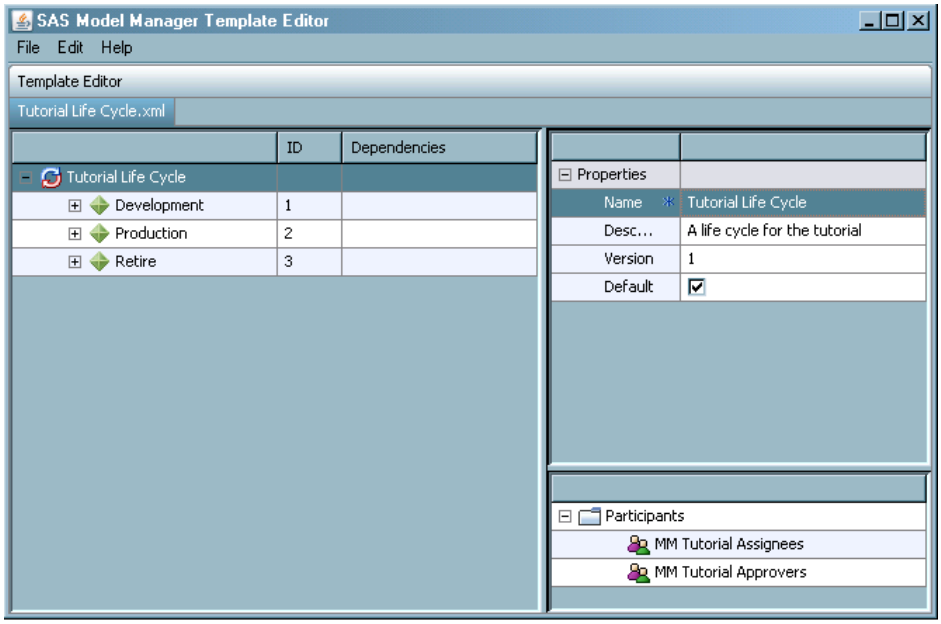
```
<LifecycleTemplate name="Tutorial Life Cycle"
   description="A life cycle for the tutorial" version="1"
   isDefault="True">
```

Rename the **mdlmgrexampleassignees** and **mdlmgrexampleapproves** participants to **`MM Tutorial Assignees`** and **`MM Tutorial Approvers`**. The participants are enclosed in <Participants> </Participants> tags. Here are the final participants in the XML file:

```
<Participants>
   <Participant id="1" name="MM Tutorial Assignees"></Participant>
   <Participant id="2" name="MM Tutorial Approvers"></Participant>
</Participants>
```

Save the file.

6. In the SAS Model Manager Template Editor, select **File** ⇨ **Open**. In the Open dialog box, select the template and click **Open**. The **Participants** list displays only **MM Tutorial Assignees** and **MM Tutorial Approvers**.




*Note:* After the correct participants have been added to the template, it is not necessary to save the template to a local computer. You can upload the template from the SAS Model Manager Template Editor. This tutorial saves the template periodically to a local computer to create a backup of the template.

## *Add a Milestone to the User-defined Template*

This exercise adds the milestone Test to the life cycle template.

1. Right-click **Tutorial Life Cycle** and select **New Milestone**. In the New Milestone window, complete the **Name** and **Type** fields and click **OK**.

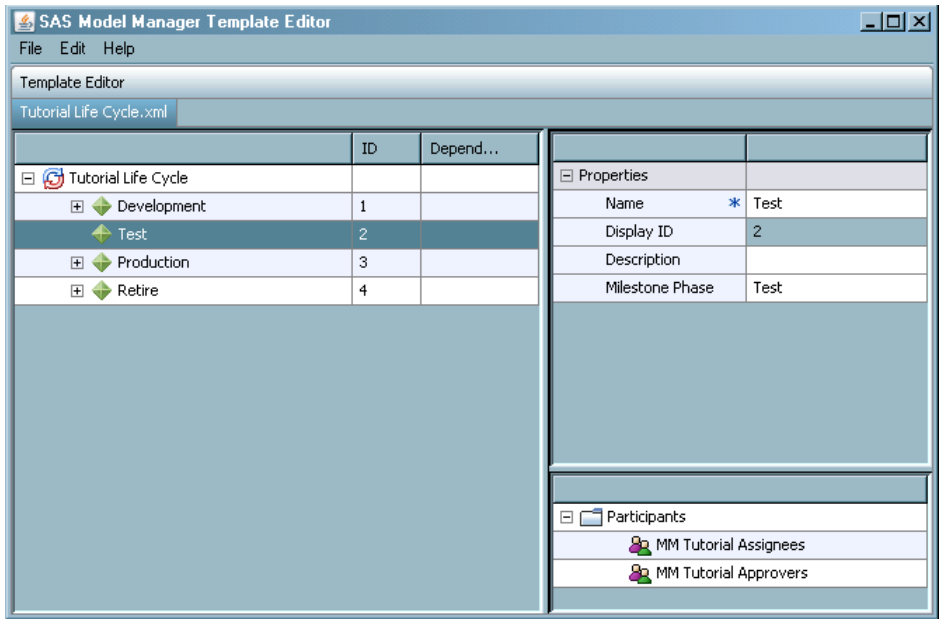
   **Name**
   enter **`Test`**.

   **Type**
   click the **Type** box and select **Test**.

   After you click **OK**, the **Test** milestone has an ID of 4.

2. Right-click **Test** and select **Move Up**. Move the **Test** milestone once more, which places it after **Development**. It now has an ID of 2.
3. Select **File** ⇨ **Save** to save the template. Click **OK** when the Warning dialog box appears.

Here is the template at the end of this exercise:




## *Add Tasks to the Life Cycle Template Milestones*

This exercise adds tasks to each milestone.

1. Add tasks to the **Development** milestone.

   For each task, right-click the **Development** milestone and select **New Task**. In the New Task window, complete the **Name** field and **Type** field using the following table. Click **OK**. The task names are descriptive. Therefore, a description is not necessary.

| Task Name | Task Type |
|---|---|
| `Define library in SAS Management Console` | **User-defined** |
| `Register data sets` | **User-defined** |
| `Set up the project in the Project Tree` | **User-defined** |
| `Import models` | **User-defined** |
| `Create comparison reports` | **User-defined** |
| `Score models` | **User-defined** |

| Task Name | Task Type |
|---|---|
| `Set the default version` | **User-defined** |
| `Sign-off` | **Sign-off** |

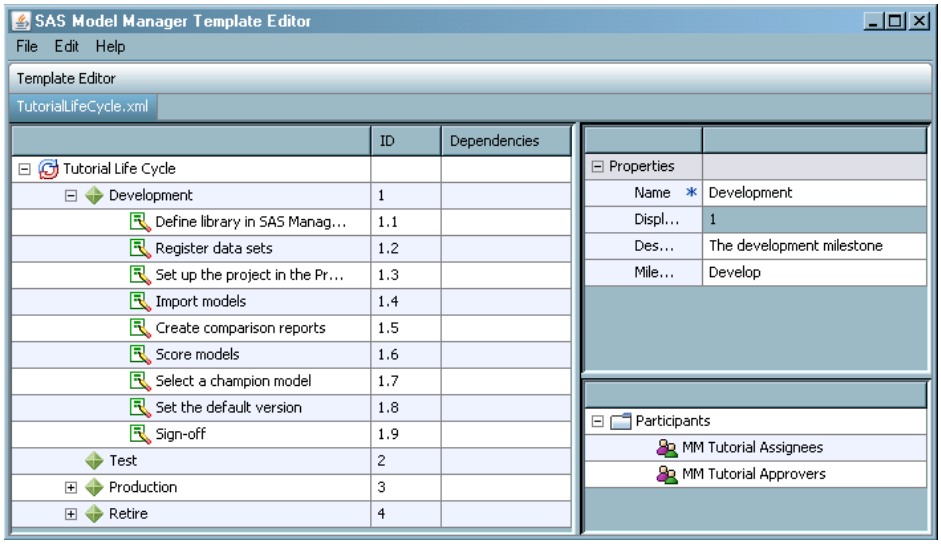
The task **Select Champion** existed in the Basic.xml sample life cycle template. Rename the task and move it after the **Score models** task:

a. In the **Name** field, change the task name to `Select a champion model`.

b. Right-click **Select Champion** and select **Move Down**. Repeat this until the task comes after **Score models**.

Expand the **Development** milestone. Each task has an ID in the form `milestone.task`. The first number in the ID is the milestone ID. The second number in the ID identifies the specific task.

Here is the template after adding the tasks for the **Development** milestone:




2. Add tasks to the **Test** milestone.

   For each task, right-click the **Test** milestone and select **New Task**. In the New Task window, complete the **Name** field and **Type** field using the following table. Click **OK**. The task names are descriptive. Therefore, a description is not necessary.

| Task Name | Task Type |
|---|---|
| `Validate score input data` | **User-defined** |
| `Validate score output data` | **User-defined** |
| `Test scoring` | **User-defined** |
| `Sign-off` | **Sign-off** |

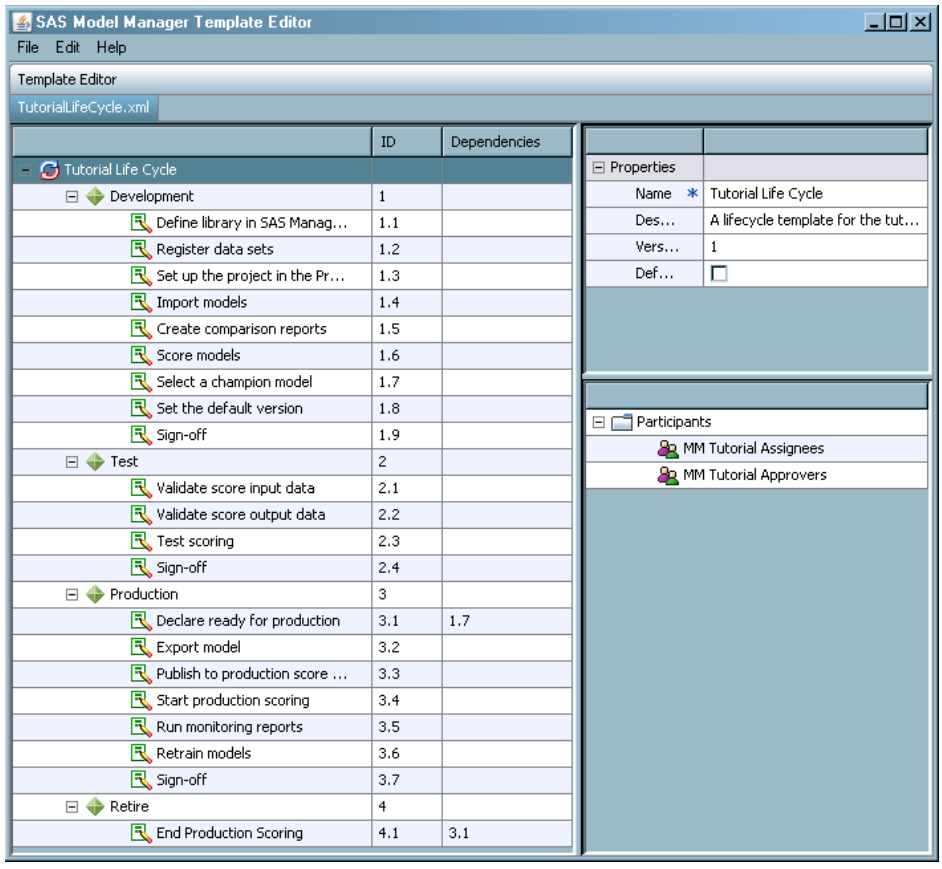
3. Add tasks to the **Production** milestone.

For each task listed below, right-click the **Production** milestone and select **New Task**. In the **New Task** window, complete the **Name** field and **Type** field using the following table. Click **OK**. The task names are descriptive. Therefore, a description is not necessary.

| Task Name | Task Type |
|---|---|
| `Export model` | **User-defined** |
| `Publish to production score server` | **User-defined** |
| `Start production scoring` | **User-defined** |
| `Run monitoring reports` | **User-defined** |
| `Retrain models` | **User-defined** |
| `Sign-off` | **Sign-off** |

4. The **Declare Ready For Production** task was provided in the Basic.xml sample template. In the **Name** field, rename the task to **Declare ready for production**.
5. Select **File** ⇨ **Save** to save the template. Click **OK** if the Warning dialog box appears.

Here is the template after all milestone tasks have been defined:



## *Add Task Dependencies*

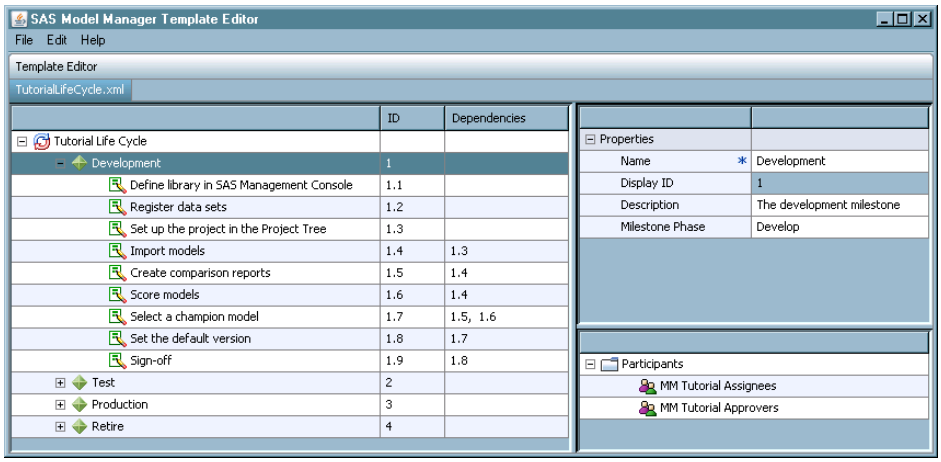
### *About Dependencies*

Your model development process might require an order to complete some tasks. To each task, you can assign dependencies that indicate that one or more tasks must be completed before another one can be marked complete. You specify dependencies in the **Dependencies** property for a task.

This exercise assigns dependencies to milestone tasks.

### *Add Dependencies for the Development Milestone Tasks*

1. Select the **Import models** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Set up the project in the Project Tree**. Click **OK**. The **Import models** task now has a dependency on task 1.3, **Set up project in the Project Tree**.
2. Select the **Create comparison reports** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Import models**. Click **OK**. The **Create comparison reports** task now has a dependency on task 1.4, **Import models**.
3. Select the **Score models** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Import models**. Click **OK**. The **Score models** task now has a dependency on task 1.4, **Import models**.
4. Select the **Select a champion model** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Create comparison reports** and **Score models**. Click **OK**. The **Select a champion model** task now has a dependency on task 1.5, **Create comparison reports** and task 1.6, **Score models**.
5. Select the **Set the default version** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Select a champion model**. Click **OK**. The **Set the default version** task now has a dependency on task 1.7, **Select a champion model**.
6. Select the **Sign-off** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Set the default version**. Click **OK**. The **Sign-off** task now has a dependency on task 1.8, **Set the default version**.
7. Here is the template after the **Development** milestone task dependencies have been assigned:

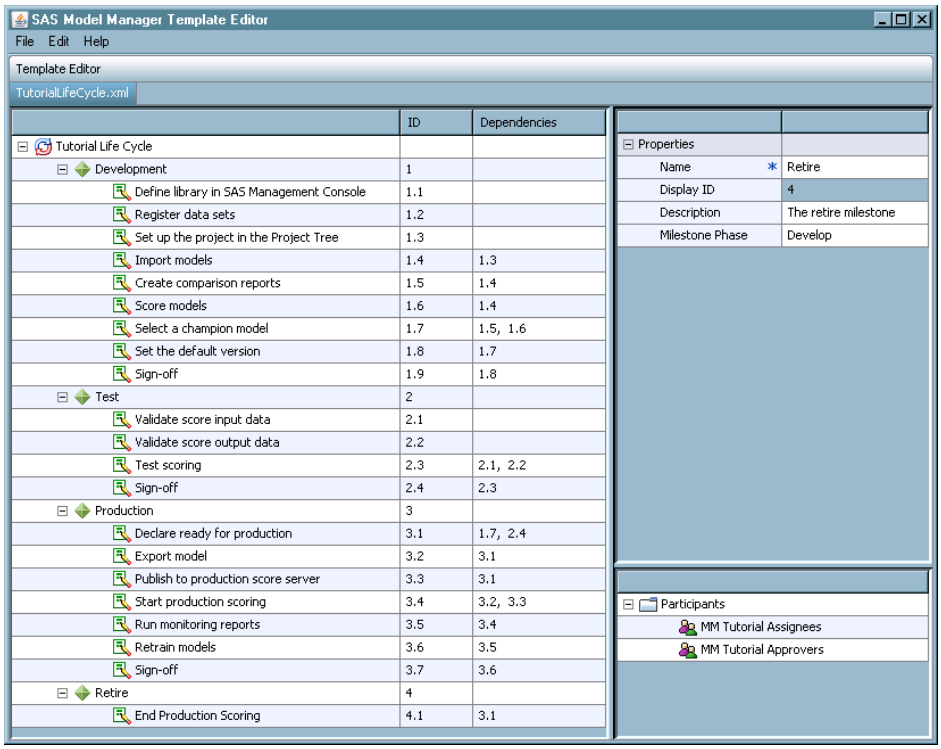


### Add Dependencies for the Test Milestone Tasks

1. Select the **Test scoring** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Validate score input data** and **Validate score output data**. Click **OK**. The **Test scoring** task now has a dependency on task 2.1, **Validate score input data**, and task 2.2, **Validate score output data**.
2. Select the **Sign-off** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Test scoring**. Click **OK**. The **Sign-off** task now has a dependency on task 2.3, **Test scoring**.

### Add Dependencies for the Production Milestone Tasks

1. Select the **Declare ready for production** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Sign-off** for **ID 2.4**. Click **OK**.
2. Select the **Export model** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Declare ready for production**. Click **OK**. The **Export model** task now has a dependency on task 3.1, **Declare ready for production**.
3. Select the **Publish to production score server** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Declare ready for production**. Click **OK**. The **Publish to production score server** task now has a dependency on task 3.1, **Declare ready for production**.
4. Select the **Start production scoring** task. Click the **Dependencies** property value field and then click the ellipsis button. Select the box for **Export model** and **Publish to production score server**. Click **OK**. The **Start production scoring** task now has a dependency on task 3.2, **Export model**, and task 3.3, **Publish to production score server**.
5. Select the **Run monitoring reports** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Start production scoring**. Click **OK**. The **Run monitoring reports** task now has a dependency on task 3.4, **Start production scoring**.
6. Select the **Retrain models** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Run monitoring reports**. Click **OK**. The **Retrain models** task now has a dependency on task 3.5, **Run monitoring reports**.
7. Select the **Sign-off** task. Click the **Dependencies** property and then click the ellipsis button. Select the box for **Retrain models**. Click **OK**. The **Sign-off** task now has a dependency on task 3.6, **Retrain models**.
8. Here is the template after all dependencies have been assigned:




9. To save the template to your local computer, select **File** ⇨ **Save**.

## *Complete Task Properties*

### *Complete the Development Task Properties*

In this exercise, you complete the **Development** task properties.

Select each task and enter property values using the values in the following table.

To assign property values for the **Assignees** and **Approvers** properties, click the ellipsis button to open the Select Participants window. Check the box for the assignee or approver and click **OK**.

| Task | Assignees Property | Approvers Property | Weight Property | Duration Property |
|---|---|---|---|---|
| Define library in SAS Management Console | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |
| Register data sets in SAS Management Console | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |
| Set up the project in the Project Tree | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |

| Task | Assignees Property | Approvers Property | Weight Property | Duration Property |
|---|---|---|---|---|
| Import models | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |
| Create comparison reports | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 2 |
| Score models | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 2 |
| Select a champion model | MM Tutorial Assignees | MM Tutorial Approvers | 20 | 4 |
| Set the default version | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |
| Sign-off | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |

Save the template.

### *Complete the Test Task Properties*

In this exercise, you complete the **Test** task properties.

Select each task and enter property values using the values in the following table.

To assign property values for the **Assignees** and **Approvers** properties, click the ellipsis button to open the Select Participants window. Check the box for the assignee or approver and click **OK**.

| Task | Assignees Property | Approvers Property | Weight Property | Duration Property |
|---|---|---|---|---|
| Validate score input table | MM Tutorial Assignees | MM Tutorial Approvers | 25 | 1 |
| Validate score output table | MM Tutorial Assignees | MM Tutorial Approvers | 25 | 1 |
| Test scoring | MM Tutorial Assignees | MM Tutorial Approvers | 40 | 5 |
| Sign-off | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |

Save the template.

### *Complete the Production Task Properties*

In this exercise, you complete the **Production** task properties.

Select each task and enter property values using the values in the following table.

To assign property values for the **Assignees** and **Approvers** properties, click the ellipsis button to open the Select Participants window. Check the box for the assignee or approver and click **OK**.

| Task | Assignees Property | Approvers Property | Weight Property | Duration Property |
|---|---|---|---|---|
| Declare ready for production | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |
| Export model | MM Tutorial Assignees | MM Tutorial Approvers | 15 | 1 |
| Publish to production scoring server | MM Tutorial Assignees | MM Tutorial Approvers | 15 | 1 |
| Start production scoring | MM Tutorial Assignees | MM Tutorial Approvers | 20 | 2 |
| Run monitoring reports | MM Tutorial Assignees | MM Tutorial Approvers | 15 | 1 |
| Retrain models | MM Tutorial Assignees | MM Tutorial Approvers | 15 | 3 |
| Sign-off | MM Tutorial Assignees | MM Tutorial Approvers | 10 | 1 |

Save the template.

## Upload the Life Cycle Template

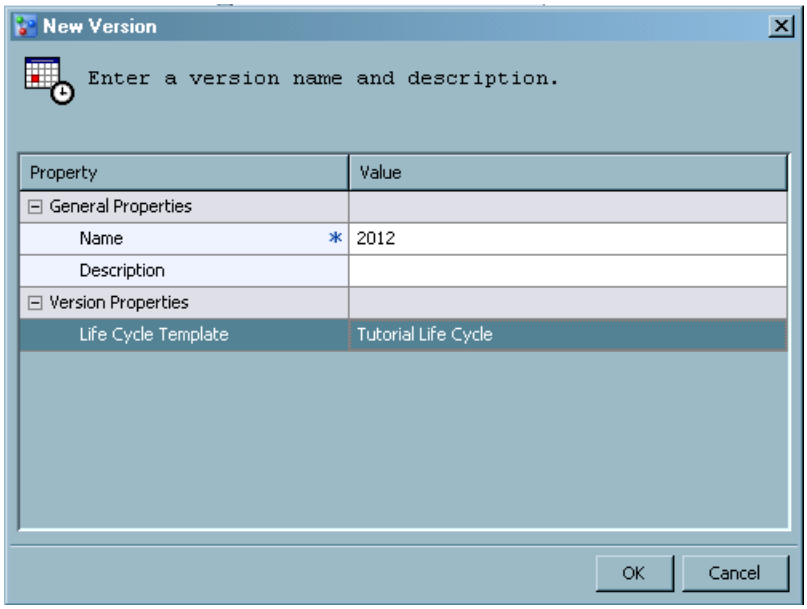
In this exercise, you upload the new life cycle template to the SAS Content Server. Only SAS Model Manager administrators can upload templates to the SAS Content Server.

1. Log on as a SAS Model Manager administrator.
2. From the SAS Model Manager windows, select **Tools** ⇨ **Manage Templates**
3. In the SAS Model Manager Template Editor, select **File** ⇨ **Open**. In the Open window, select TutorialLifeCycle.xml and click **OK**.
4. Select **File** ⇨ **Upload File**. Verify the filename in the Upload File window and click **OK**. A message box appears when the file was uploaded successfully.
5. You can now view this life cycle template in the Browse Templates window and in the Life Cycle perspective.

   To view the template in the Browse Templates window, select **File** ⇨ **Browse** ⇨ **Browse Templates** ⇨ **TutorialLifeCycle.xml** and click **Open**.

   To view the template in the Life Cycle perspective, in the SAS Model Manager window, select the Life Cycle perspective button  .

**Tutorial Life Cycle** can now be specified as a life cycle template, as shown in the New Version window, when you create a version in subsequent tutorials:




# Create a Workflow Process Instance

## Overview

An *instance* is a working version of a workflow process definition. Only a SAS Model Manager administrator can create a new workflow instance. Each workflow instance consists of activities. Activities can contain user-defined properties and comments so that you can share information with other users, or make notes. The status that you select when completing an activity determines the next activity in the workflow process.

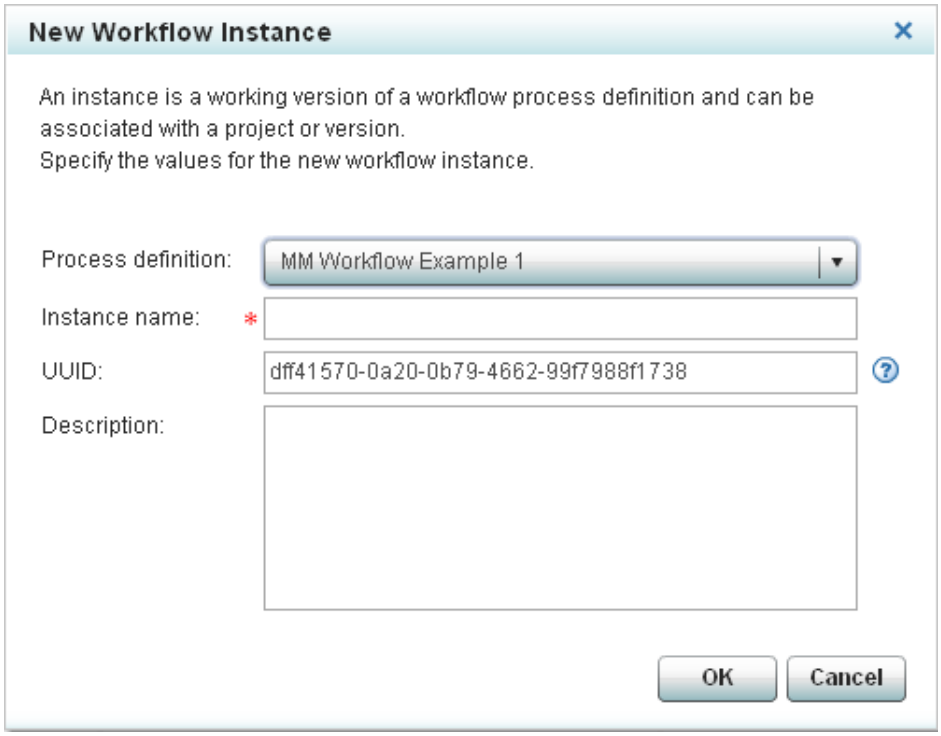
## Prerequisites

The exercises in this tutorial require that you have made the workflow process definition available to SAS Model Manager. For more information, see "Prepare for Using SAS Workflow" on page 13.

## Create a New Workflow Instance

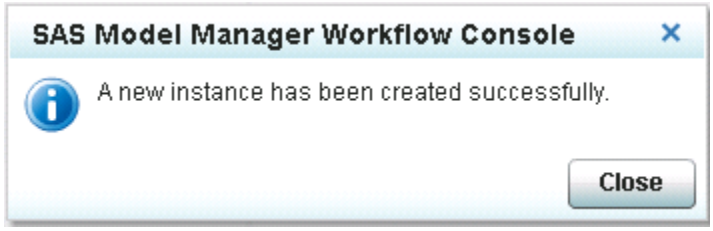
1. Log on to SAS Model Manager as a member of the **Model Manager Administrator Users** group.
2. From the SAS Model Manager main window, right-click a project or version and select **New Workflow Instance**. Workflow Console is launched in a Web browser and displays the New Workflow Instance window.

   *Note:* If you are already logged on to Workflow Console, from the Process Definitions category view, select a process definition and click .

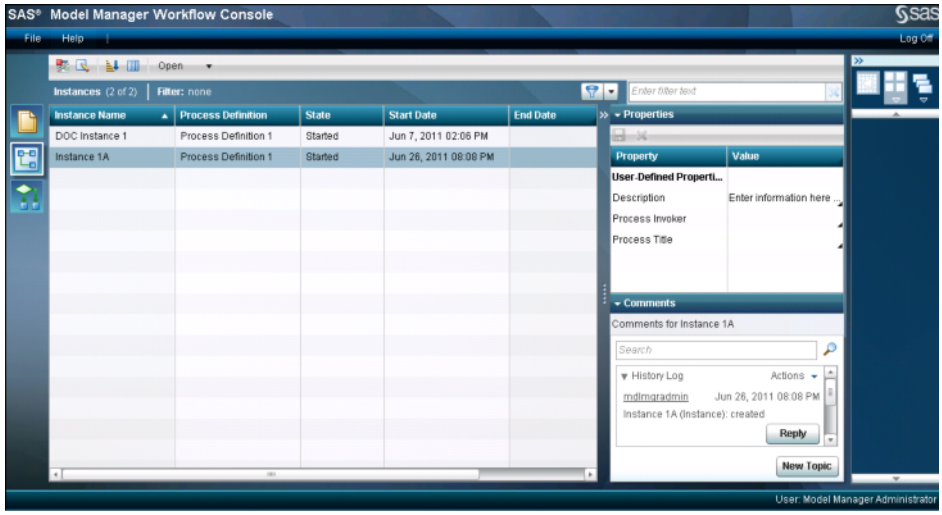



3. Select the process definition associated with the tutorial for which that you are creating the instance, if you accessed the New Workflow Instance window from the SAS Model Manager main window.
4. Enter a name for the instance.
5. The UUID of the selected project or version is already populated.

   *Note:* If the UUID is not already populated, you can copy the UUID system property value for a project or version from the Properties view in the SAS Model Manager main window. The field label and other characters that precede the UUID value must be removed.
6. (Optional) Enter a description for the instance.
7. Click **OK**. A message appears, indicating that the instance has been successfully created.




8. Click **Close**. The new workflow instance is now available in the Instances category view.
9. To view the new instance, click  . The Instances category view appears. Select the instance to view information that is associated with the new instance.




The workflow process definitions that have been provided for the tutorials already has participants assigned. For information about how to assign additional participants to an instance of a workflow process definition, see “Working with Workflow Participants” in Chapter 7 of *SAS Model Manager: User's Guide*. You can also see Chapter 11, “Tutorial 10: Using Workflow Console,” on page 165 to learn how to manage workflow instances and work with activities.

*Chapter 3*

# Tutorial 2: Performing Basic SAS Model Manager Tasks

# Overview of SAS Model Manager Basics

After an administrator has defined your user ID in SAS Management Console and you have a life cycle template, you are ready to work in SAS Model Manager. This tutorial guides you through a simple modeling project process.

To enable you to track the progress of the modeling project, each task has instructions to update the version life cycle status.

# Prerequisites

### *Tutorial 2 Models and Data Sets*

The exercises in this tutorial require that the Tutorial 2 data sets and models from SMM31Tutorial.zip be extracted and registered in SAS Management Console. If they have not been extracted and registered, see "Prepare Tutorial 2 Data Sets and Models" on page 4 to extract and register the files. You must have access from the SAS Model Manager client to the tutorial files.

### *Verify Your User ID as a Member of Model Manager User Groups*

This exercise ensures that your user ID is a member of the **MM Tutorial Assignees** group and the **Model Manager Advanced Users** group.

1. Open SAS Management Console and log on to the SAS Metadata Server.
2. On the **Plug-ins** tab, select **User Manager**.
3. In the right pane, double-click the **MM Tutorial Assignees** group and click the **Members** tab.
4. Review the **Current Members** list, and locate your user ID or a group that your user ID is a member of. If your user ID or group is not a member of the **MM Tutorial Assignees** group, ask your administrator to add you to this group. Close the properties window.
5. Find and double-click your user ID in the right pane of SAS Management Console.
6. Click the **Groups and Roles** tab. Review the **Member of** pane and locate the group **Model Manager Advanced Users**. If your user ID is not a member of this group, ask your administrator to add you to this group. Close the properties window.
7. Close SAS Management Console.

### *See Also*

"Create Groups for Use with the SAS Model Manager Tutorial" on page 20

# Organize the Model Hierarchy

In this exercise, you learn to use the Project Tree to create a modeling project.

### Create a Folder

To provide an organizational folder to manage your modeling projects, follow these steps:

1. Click the Project perspective button  and right-click **MMRoot** in the Project Tree. Select **New** ⇨ **New Folder**. The New Folder dialog box appears.
2. Specify values for the following folder properties and click **OK**.

   **Name**
   enter `Tutorial2` for the folder name.

   **Description**
   enter an optional folder description.

The new folder appears in the Project Tree.

### Create a New Project

To create a project and define the model function, follow these steps:

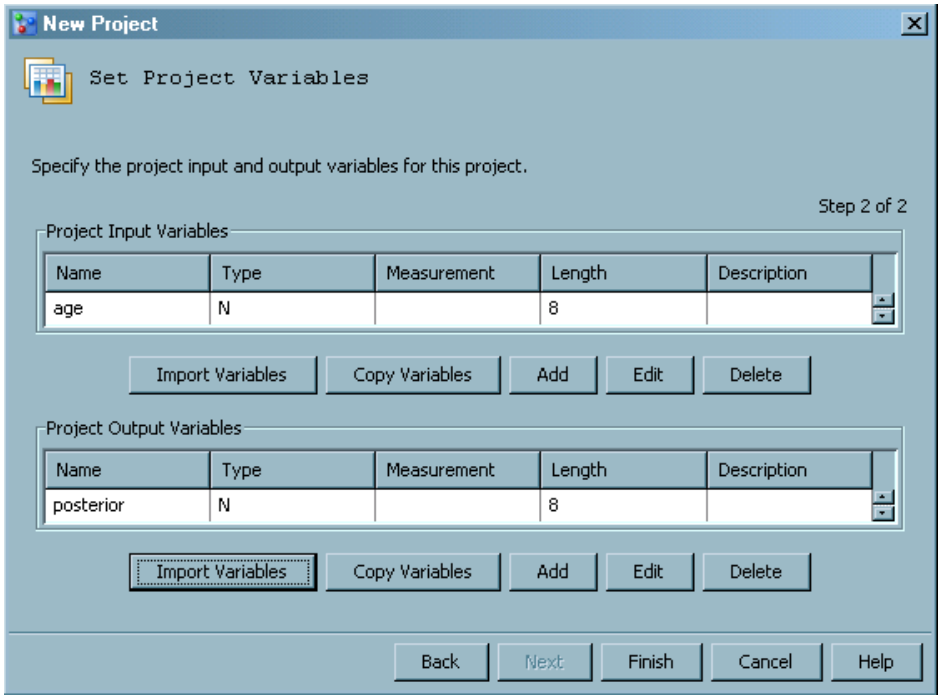
1. Right-click **Tutorial2** and select ⇨ **New** ⇨ **New Project**. The New Project wizard appears.
2. Specify the following project properties and click **Next**:

   **Name**
   enter `Delinquency` for the project name.

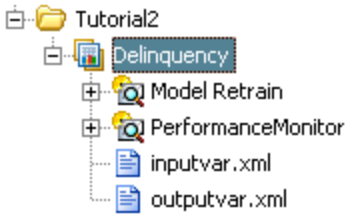
   **Description**
   enter an optional description.

   **Model Function**
   select **classification**.
3. Specify the project input variables:
   a. Below the **Project Input Variables** table, click **Import Variables**. The Import Variables from Table window appears.
   b. Double-click these folders: **Shared Data** ⇨ **Model Manager** ⇨ **Tutorial2**.
   c. Select **DELINQUENCY_PROJECT_INPUT** and click **OK**.
4. Specify the project output variables:
   a. Below the **Project Output Variables** table, click **Import Variables**. The Import Variables from Table window appears.
   b. Select **DELINQUENCY_PROJECT_OUTPUT**, click **OK**, and click **Finish**.

New Project Wizard Step 2 of 2 after the project variables have been set.




5. Examine the **Tutorial2** folder to verify that it contains the **Delinquency** project.




### *Define the Project Properties*

To define the properties that SAS Model Manager uses to create reports and score models, follow these steps:

1. Select the **Delinquency** project in the **Tutorial2** folder and expand **Specific Properties** in the right pane.
2. Specify the default data tables and model variables for the project, right-click **Delinquency**.

   **Default Test Table**
   select **DELINQUENCY_TEST**.

   **Default Scoring Task Input Table**
   select **DELINQUENCY_SCORING_INPUT**.

   **Default Scoring Task Output Table**
   select **DELINQUENCY_SCORING_OUTPUT**.

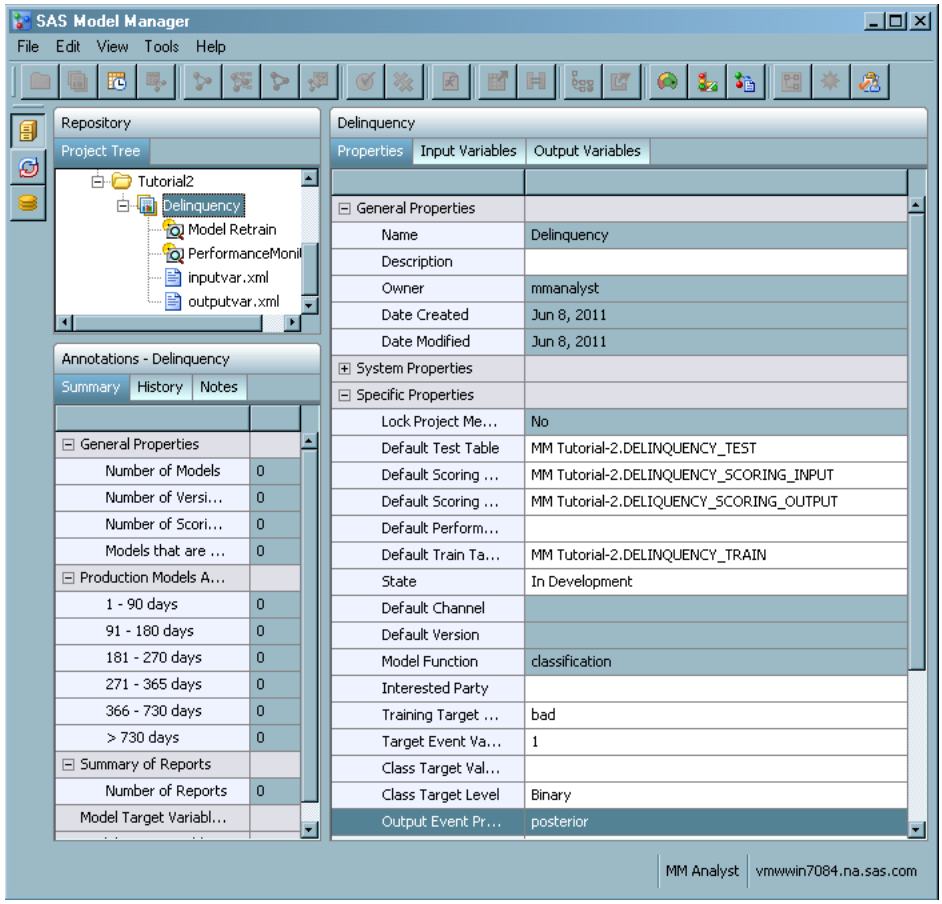
   **Default Train Table**
   select **DELINQUENCY_TRAIN**.

**Training Target Variable**
enter **bad**.

**Target Event Value**
enter **1**.

**Class Target Level**
select **Binary**.

**Output Event Probability Variable**
select **posterior**.




## *Create a Version*

Create a version for the project. The version folder contains life cycle information, auxiliary version documents, candidate model files, model comparison reports, resource files, scoring tasks, and model performance reports. To create a new version, follow these steps:

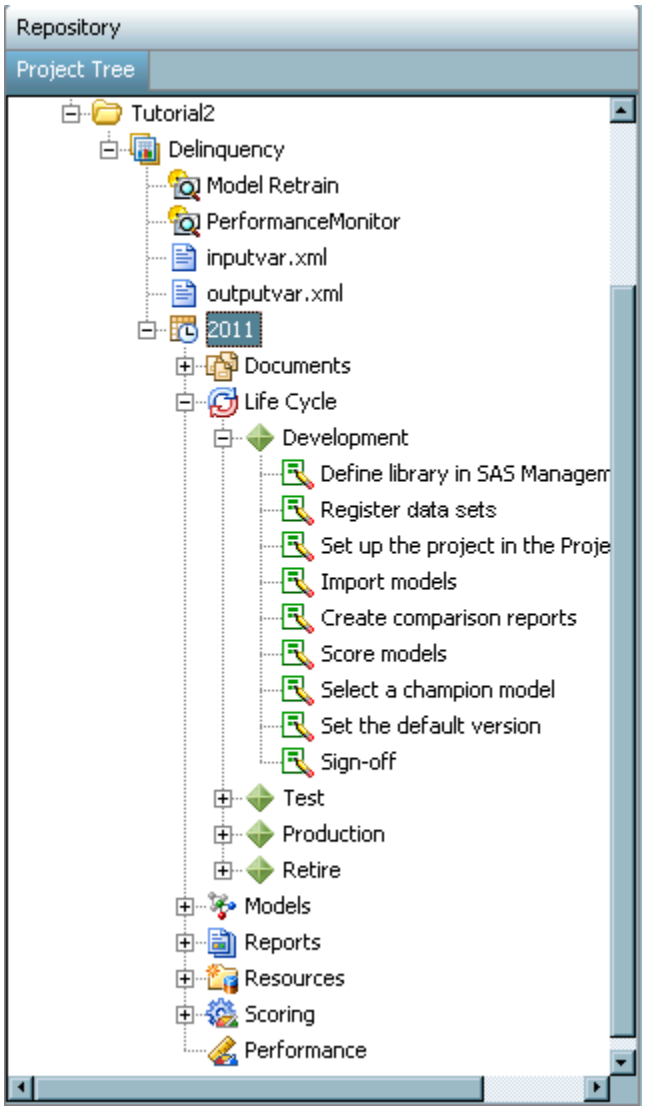
1. Right-click the **Delinquency** project and select **New** ⇨ **New Version**. The New Version dialog box appears.
2. Specify the following version properties and click **OK**.

   **Name**
   enter **2011** for the version name.

   **Life Cycle Template**
   select the user-defined template **Tutorial Life Cycle**.

*Note:* If you are using a workflow process to track the progress of your project or version, you can select any life cycle template. You can then skip all tasks to update the life cycle.

3. Examine the **Delinquency** project to verify that it contains one version called **2011**. Select **Life Cycle**. Verify that the **Name** property is **Tutorial Life Cycle**.

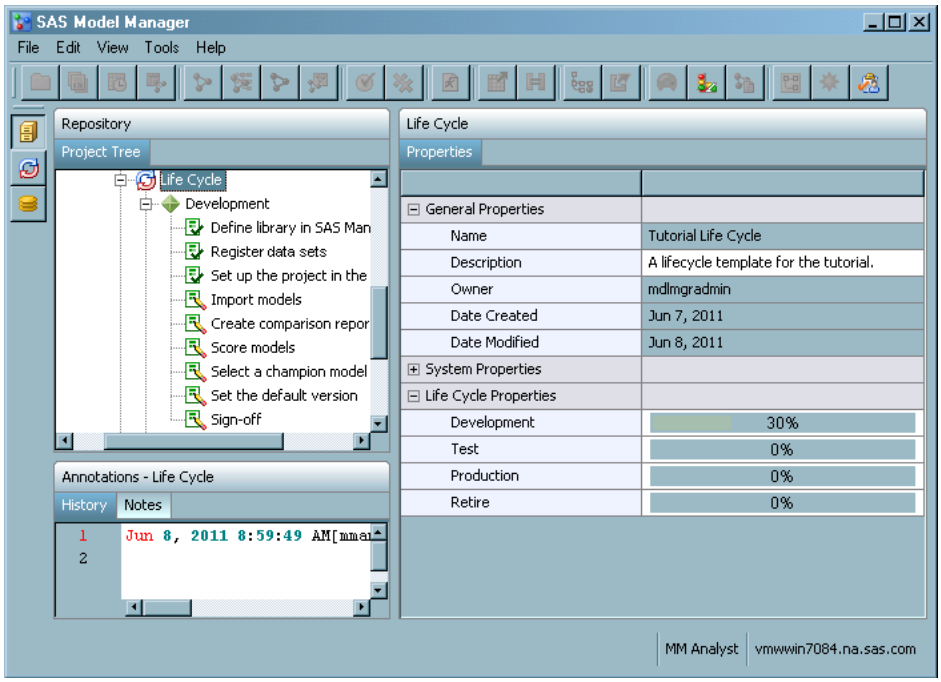



*Note:* If you want to use a workflow process to track the progress of your project or version, send a request to a SAS Model Manager administrator and ask the administrator to create an instance of a workflow process definition to use for the tutorials. Include the name and UUID of the project or version with which you want to workflow process instance to be associated.

### Update the Life Cycle (Optional)

To complete the milestone task of adding data sources and setting up the project in the Project Tree, follow these steps:

1. In the **Delinquency** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Define library in SAS Management Console** task and examine the task properties. The **To Be Completed By** property, assigned in the life cycle template, determines which users or user groups from the Participants list are responsible for this milestone task. Because you are a member of the **MM Tutorial Assignees** group, you are authorized to update the task status.
3. Click the **Status** box and select **Completed**.
4. Select the **Register data sets** task and examine the task properties. Click the **Status** box and select **Completed**.
5. Select the **Set up project in the Project Tree** task. Click the **Status** box and select **Completed**.
6. Select all of the tasks whose status you updated and examine the properties. Verify that the value of the **Completed Date** property is today and that the value of the **Completed By** property is your user ID.
7. Select the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.




### Update the Workflow Process (Optional)

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

# Import Models

In this exercise you import models into SAS Model Manager, set model properties, and map the model variables. The imported models are SAS code models. The SAS code for the first model is the LOGISTICS procedure, whereas the SAS code for the second and third models consists of DATA step fragments. To import SAS code file, at least three component files are required: the model score code, the model input file, and the model output file. For prediction or classification models, you also must prepare model target files.

## *Import SAS Code Models*

1. Expand the **2011** version in the **Delinquency** project and right-click **Models** folder. Then select **Import from** ⇨ **Local Files**. The Local Files window appears.

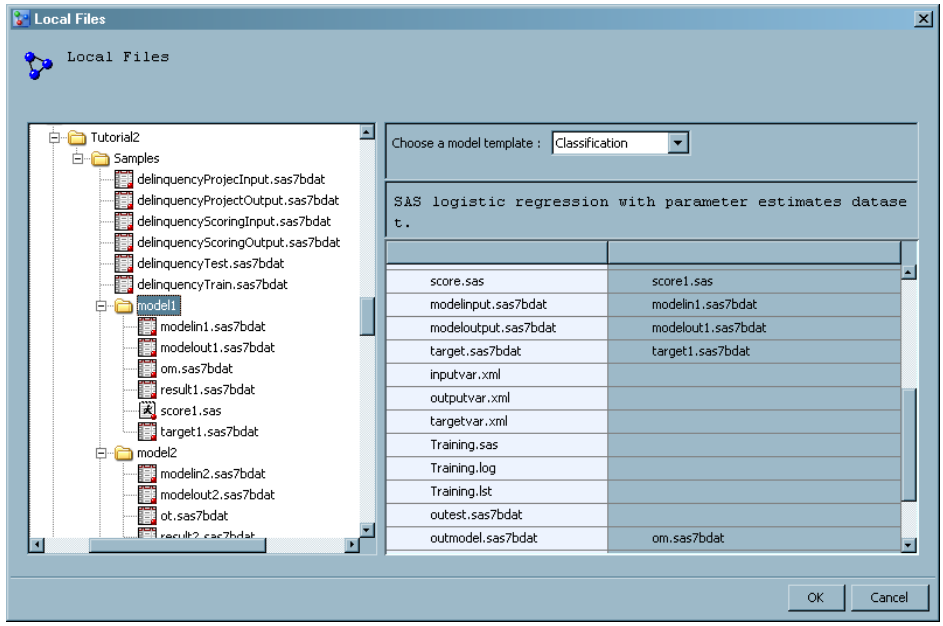
2. Import Model 1.

   a. In the left pane, expand the **Desktop** folder and select **`<drive>:\Tutorial2\Samples\model1`**.

   b. Type **`Model 1`** in the **Name** box. For each filename in the Object column, click the filename and drag it to the corresponding option box. This action maps the tutorial model component filenames to the SAS Model Manager model component filenames.

| Object | Option |
|---|---|
| **modelin1.sas7bdat** | **modelinput.sas7bdat** |
| **modelout1.sas7bdat** | **modeloutput.sas7bdat** |
| **om.sas7bdat** | **outmodel.sas7bdat** |
| **score1.sas** | **score.sas** |
| **target1.sas7bdat** | **target.sas7bdat** |

Here is the Local Files window after the files have been mapped.




c. Click **OK**.

3. Import Model 2.

   a. Open the Local Files. In the left pane, expand the **Desktop** folder and select `<drive>:\Tutorial2\Samples\model2`.

   b. Type `Model 2` in the **Name** box. For each filename in the Object column, click the filename and drag it to the corresponding option box. This action maps the tutorial model component filenames to the SAS Model Manager model component filenames.

| Object | Option |
|---|---|
| **modelin2.sas7bdat** | **modelinput.sas7bdat** |
| **modelout2.sas7bdat** | **modeloutput.sas7bdat** |
| **ot.sas7bdat** | **outmodel.sas7bdat** |
| **score2.sas** | **score.sas** |

| Object | Option |
|---|---|
| **target2.sas7bdat** | **target.sas7bdat** |

c. Click **OK**.

4. Import Model 3.

   a. Open the Local Files window. In the left pane, expand the **Desktop** folder and select **`<drive>:\Tutorial2\Samples\model3`**.

   b. Type **`Model 3`** in the **Name** box. For each filename in the Object column, click the filename and drag it to the corresponding option box. This action maps the tutorial model component filenames to the SAS Model Manager model component filenames.

| Object | Option |
|---|---|
| **modelin3.sas7bdat** | **modelinput.sas7bdat** |
| **modelout3.sas7bdat** | **modeloutput.sas7bdat** |
| **score3.sas** | **score.sas** |
| **target3.sas7bdat** | **target.sas7bdat** |

   c. Click **OK**.

5. Examine the **Models** folder to verify that it contains the three models. Right-click the folder and select **Expand All Items** to examine the model files.

### *Set Model Properties*

Set the properties for the model. SAS Model Manager requires that the **Score Code Type** be set to **Data Step** if the score code is a DATA step fragment, or be set to **SAS Program** if the score code is a SAS procedure. Follow these steps.

1. Select **Model 1**. Click the **Description** field and enter **`first model for tutorial 2`**.
2. Select **Model 2**. Click the **Score Code Type** box and select **DATA Step**.
3. Select **Model 3**. Click the **Score Code Type** box and select **DATA Step**.

### *Map Model Variables to Project Variables*

When the names of the model output variable are not identical to the names of the project output variables, you must map the variables. To map model output variables to project output variables, follow these steps:

1. Map model variables for the first model. Right-click **Model 1** in the **Models** folder and then select **Set Model Output Mapping**. Ensure that the following model variables are mapped to their respective project variables. To map a model variable

to a project variable, click in the box in the **Model Variables** column, select a variable, and click **OK**.

| Project Variables | Model Variables |
|---|---|
| **posterior** | **P_1** |
| **prediction** | **I_bad** |

2. Map model variables for the second model. Select **Model 2** in the **Models** folder and then click the **Model Mapping** tab in the right pane. Click **Edit**. Ensure that the following model variables are mapped to their respective project variables. To map a model variable to a project variable, click the box in the **Model Variables** column and select a variable. Click **OK** when you are finished.

| Project Variables | Model Variables |
|---|---|
| **posterior** | **prob2** |
| **prediction** | **prediction** |

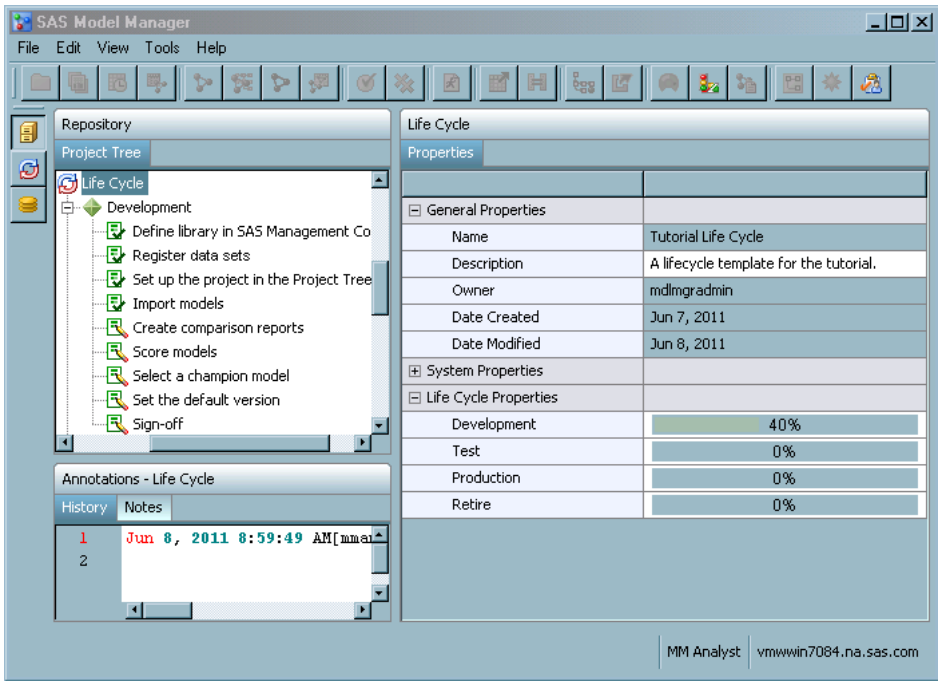
3. Map model variables for the third model. Select **Model 3** in the **Models** folder and then click the **Model Mapping** tab in the right pane. Click **Edit**. Ensure that the following model variables are mapped to their respective project variables. To map a model variable to a project variable, click the box in the **Model Variables** column and select a variable. Click **OK** when you are finished.

| Project Variables | Model Variables |
|---|---|
| **posterior** | **p_bad1** |
| **prediction** | **prediction** |

### *Update the Life Cycle (Optional)*

To update the Development milestone, follow these steps:

1. In the **Delinquency** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Import models** task. Click the **Status** box and select **Completed**.
3. Select the **Development** milestone to refresh the property values. Select **Import models**. The **Completed Date** and **Completed By** fields have been updated with today's date and your user ID.
4. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Develop** property displays a bar chart that shows the percentage of completed tasks for this milestone.




### *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, “Tutorial 10: Using Workflow Console,” on page 165.

# Create Model Comparison Reports

In this exercise, you create several model comparison reports that are used in the selection and approval of a champion model. After you create the reports, you view the reports in the **Reports** folder. The reports enable you to evaluate candidate models in a version or across versions by assessing the structure, performance, and resilience of your models.

## *Create a Model Profile Report*

The Model Profile report creates three tables to display the profile data that is associated with the model input variables, output variables, and target variables. To create this report, follow these steps:

1. Expand the **2011** version in the **Delinquency** project and right-click the **Reports** folder. Then select **Reports** ⇨ **New Report Wizard**. The New Report Wizard opens.
2. In the New Report Wizard, use the specified values for these fields and click **OK**:

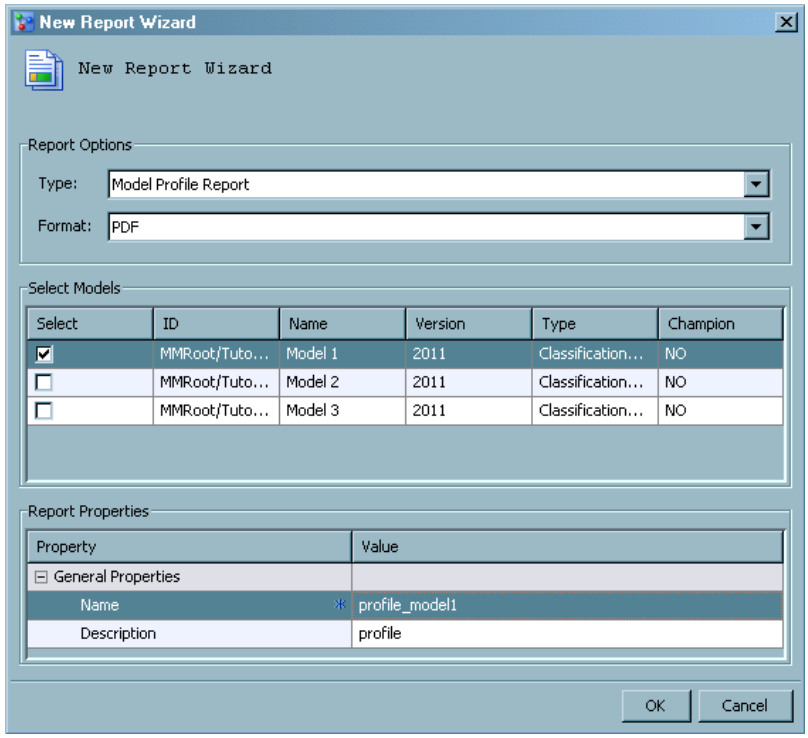
   **Type**
   select **Model Profile Report**.

   **Format**
   select **PDF**. **PDF** is the default value, and it might already be the value for **Format**.

   **Select Models**
   select the box for **Model 1**.

   **Report Properties**
   replace the default report name with the report name **profile_model1** in the **Name** field.

   Here is the New Report Wizard at this point in the process. Click **OK** when you are finished.




3. When the information dialog box confirms that the report was created successfully, click **Close**.

## *Create a Delta Report*

The Delta report compares the profile data for two models and notes the differences. To create this report, follow these steps:

1. Expand the **2011** version in the **Delinquency** project and right-click the **Reports** folder. Then select **Reports** ⇨ **New Report Wizard**. The New Report Wizard opens.

2. In the New Report Wizard, use the specified values for these fields and click **OK**:

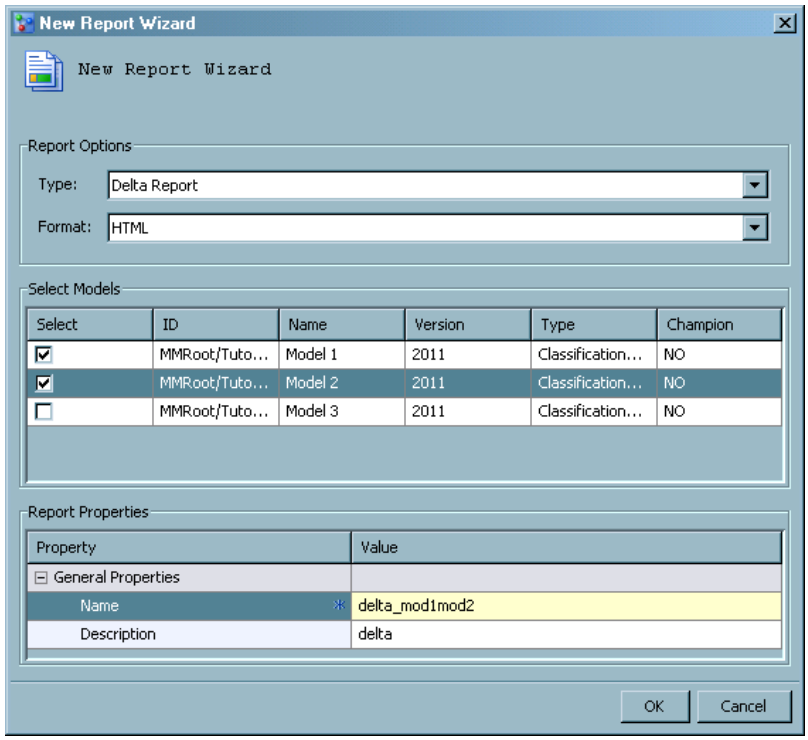
   **Type**
   select **Delta Report**.

   **Format**
   select **HTML**.

   **Select Models**
   select the boxes for **Model 1** and **Model 2**.

   **Report Properties**
   replace the default report name with the report name **delta_mod1mod2** in the **Name** field.

   Here is the New Report Wizard at this point in the process. Click **OK** when you are finished.




3. When the information dialog box confirms that the report was created successfully, click **Close**.

## *Create a Dynamic Lift Report*

The Dynamic Lift report provides visual summaries of the performance of one or more models for predicting a binary outcome variable performance. To create this report, follow these steps:

1. Expand the **2011** version in the **Delinquency** project and right-click the **Reports** folder. Then select **Reports** ⇨ **New Report Wizard**. The New Report Wizard opens.

2. In the New Report Wizard, specify the following options and click **OK**:

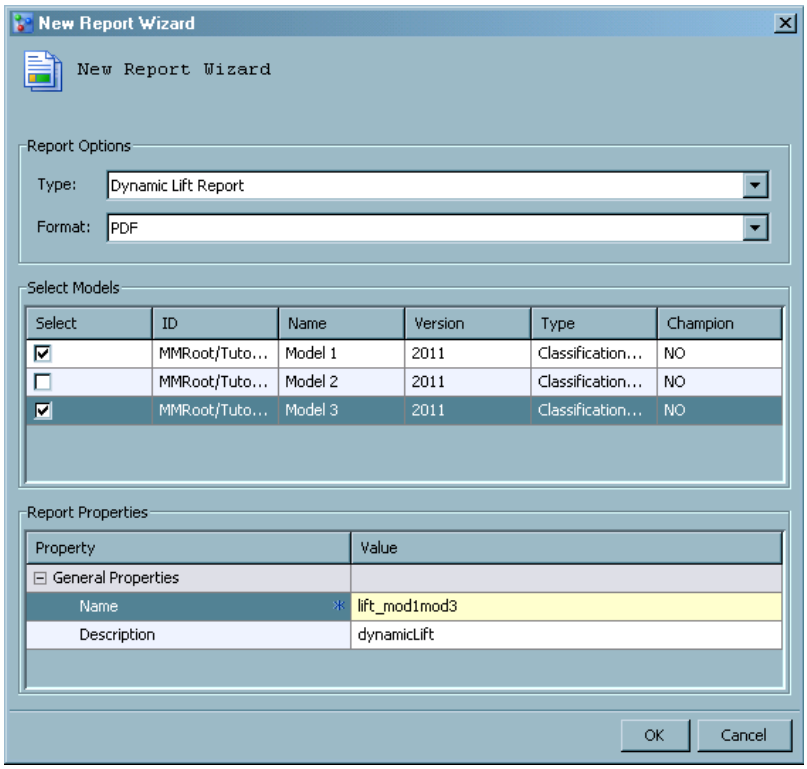
   **Type**
   select **Dynamic Lift Report**.

   **Format**
   select **PDF**.

   **Select Model(s)**
   select the boxes for **Model 1** and **Model 3**.

   **Report Properties**
   replace the default report name with the report name **lift_mod1mod3** in the **Name** field.

   Here is the New Report Wizard at this point in the process. Click **OK** when you are finished.




3. When the information dialog box confirms that the report was created successfully, click **Close**.

## *View a Model Comparison Report*

To view a model comparison report, follow these steps:

1. Expand the version folder **2011** and the **Reports** folder.
2. Right-click the report name and select **Reports** ⇨ **View Report**.

   *Note:* If user credentials are required, then specify a user ID and password that have permission to access the SAS Content Server.
3. Use the PDF viewer to distribute or print a copy of the report. In Adobe Reader, select **File** ⇨ **Send** ⇨ **Page by E-mail**.
4. Close the PDF Viewer.

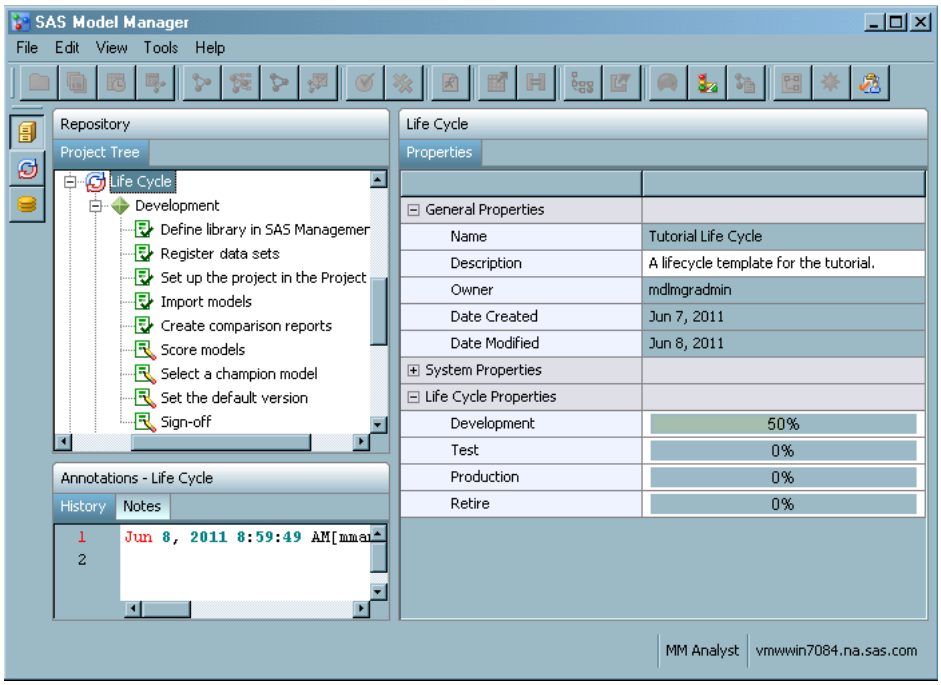
For a detailed description of the model comparison reports, see the *SAS Model Manager: User's Guide*.

## *Update the Life Cycle (Optional)*

To update the Development milestone, follow these steps:

1. In the **Delinquency** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Create comparison reports** task. Select the **Status** box and select **Completed**.
3. Select **Create comparison reports**. The **Completed Date** and **Completed By**fields have been updated with today's date and your user ID.

4. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.




## *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

# Using the Annotation View

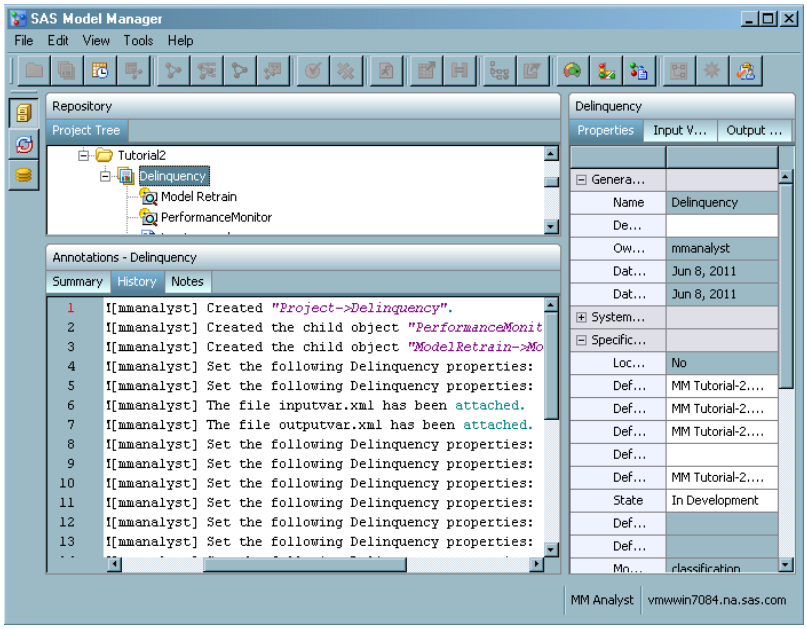
In this exercise, you use the **Annotations** view to examine the time-stamped event log, add text information to a model component, and examine the synopsis of the project and model reports. The Annotations view is the lower left pane of the SAS Model Manager window.

## *View History*

To view the event log for the different components of a project, follow these steps:

1. Select the **Delinquency** project.
2. In the **Annotations** view, click the **History** tab. This tab displays a time-stamped entry each time you create, modify, import, publish, export, or delete a component. SAS Model Manager records the following information:
   - The date and time that the action occurred
   - The user ID that performed the action
   - The action that was performed

Here is an example of the **History** in the **Annotations** view:

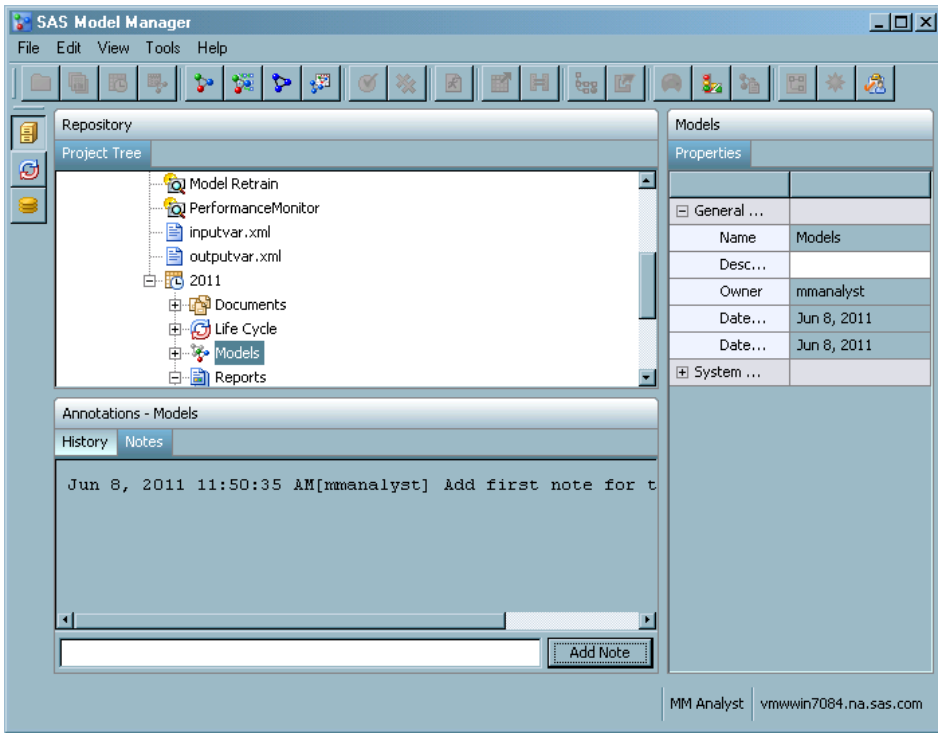



## *View Notes*

To create persistent annotations that are associated with the different components of a project, follow these steps:

1. Expand the **2011** version in the **Delinquency** project and select the **Models** folder.
2. In the **Annotations** view, click the **Notes** tab.
3. In the **Add Notes** field, enter `Add first note for tutorial 2 models` and click **Add Notes**.

   Here is the Delinquency project note in the **Annotations** view:



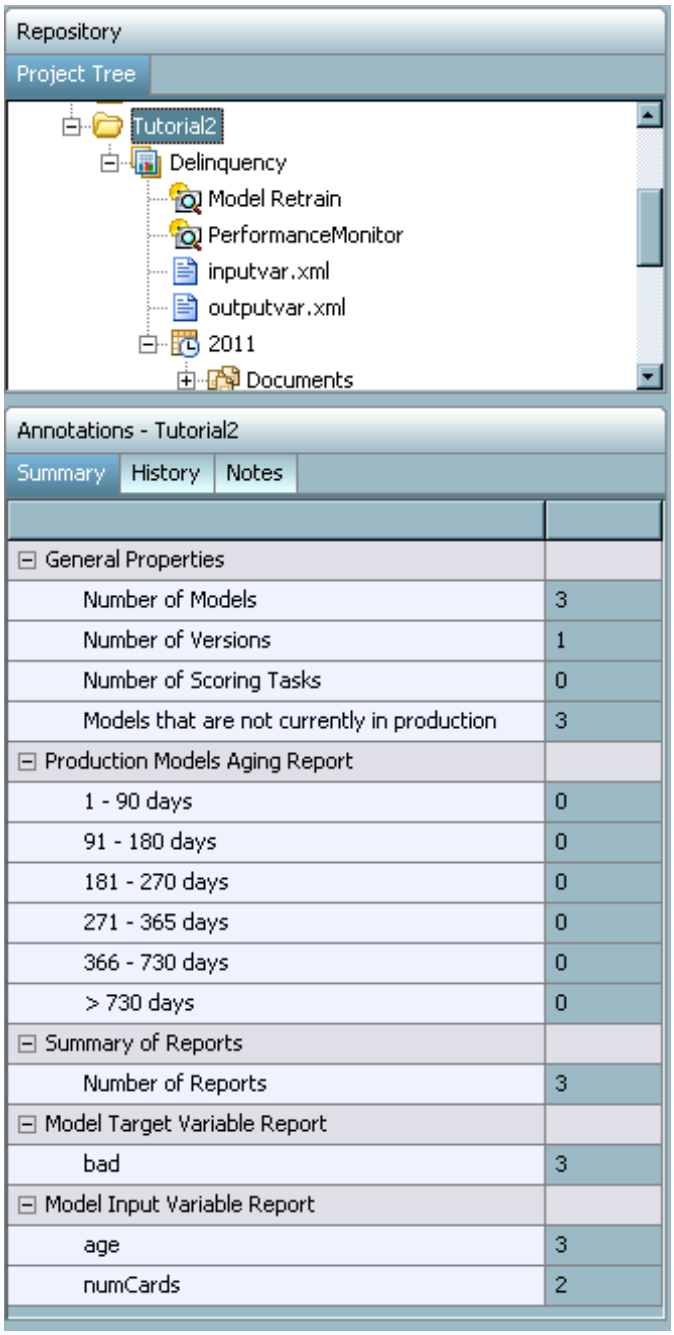

### *View the Summary Information*

The Summary contains information about the components that are contained in the selected folder.

To view summary information, follow these steps:

1. In the Project Tree, click the **Tutorial2** folder.
2. In the **Annotations** view, click the **Summary** tab.
3. Examine the information on this tab. SAS Model Manager includes general property information about the components that are part of this folder, aging information, number of reports, target information, and input information about the project.

Here is the **Summary** information for **Tutorial2**:




The information on the **Summary** tab dynamically reflects the contents of the selected node and its subnodes in the Project Tree. If you select **MMRoot**, you see summary information for all nodes in the Project Tree. If you select a project, the summary information reflects the project and all nodes within that project.

## Scoring Models

In this exercise, you create a scoring task that is used to run the score code of a model and produce scoring results. You use the results to determine the scoring accuracy and to

analyze the model performance. The scoring task uses data from a scoring task input table, and then generates the results in a scoring task output table.

### *Create a Scoring Task*

1. Expand the **2011** version, right-click the **Scoring** folder and select **New Scoring Task**. The New Scoring Task wizard appears.
2. Specify the following options and click **Next**:

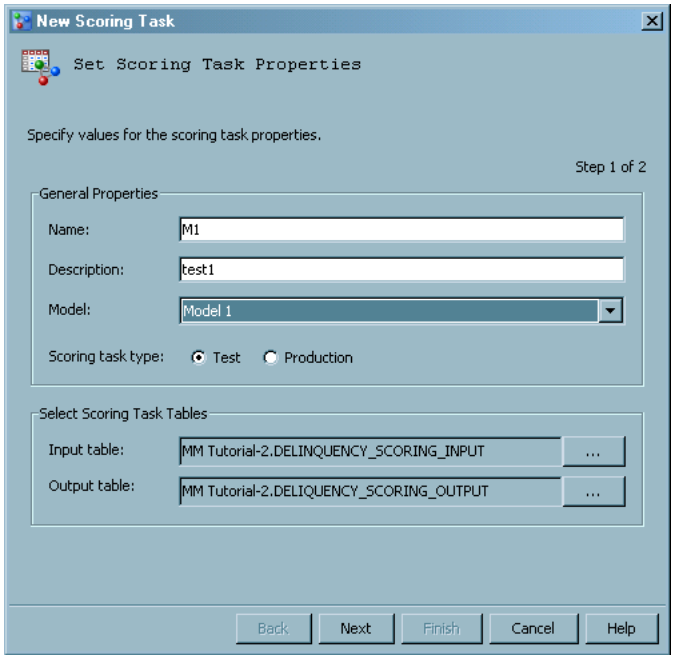
   **Name**
   enter `M1` for the scoring task name.

   **Description**
   enter `test1`.

   **Model**
   select **Model 1**. This model controls the available values for the input and output tables.

   **Scoring task type**
   select **Test**.

   *TIP* A best practice is to start all scoring tasks with **Test** selected. The results of a scoring task when it is run as type **Test** are not overwritten. You can change the type to **Production** after you are satisfied with the scoring task results and when the model is ready for production.




3. Verify that the output variables are mapped to the model variables. The variable mapping is as follows:

| Output Variable | Model Variable |
|---|---|
| P_0 | P_0 |
| posterior | P_1 |

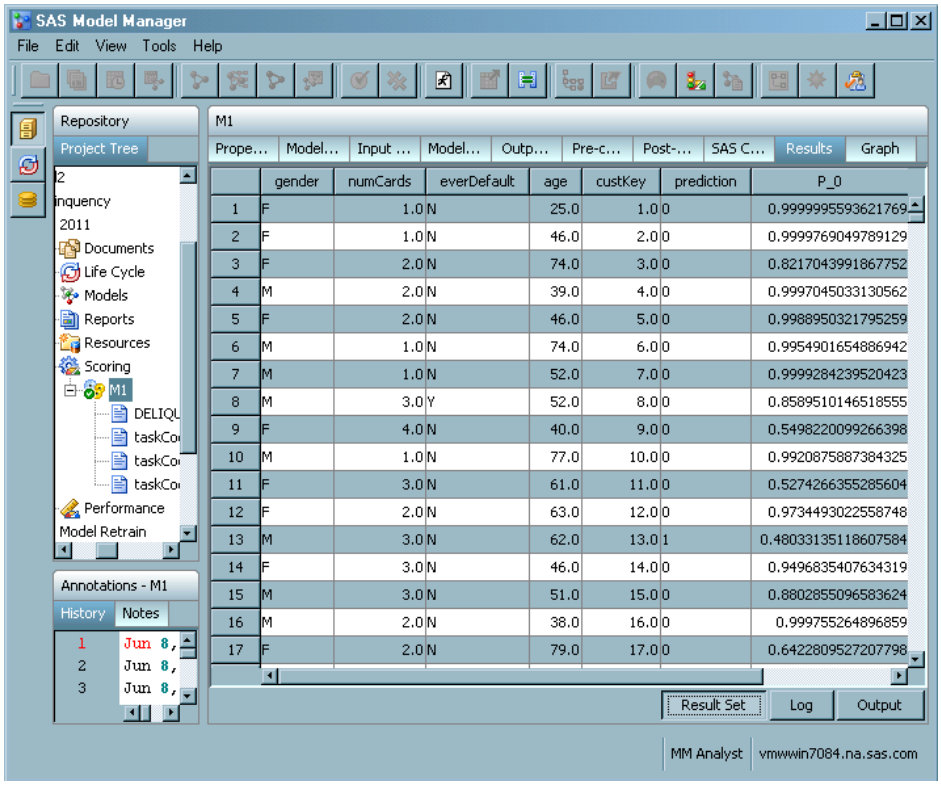
| Output Variable | Model Variable |
|---|---|
| age | age |
| prediction | I_bad |
| gender | gender |
| custKey | custKey |
| numCards | numCards |
| everDefault | everDefault |

Click **Finish**.

4. Select the **M1** scoring task to examine its properties. The value for **Date Modified** is today's date. To change the scoring task name or model input and output tables, you must create a new scoring task.

### *Execute a Scoring Task*

1. Verify that you have mapped the model output variables to the scoring task output variables. For more information, see "Map Model Variables to Project Variables" on page 47.
2. Validate the input variables. Expand the **Scoring** folder, select the **M1** scoring task and click the  toolbar button. Examine the results of **Quick Check**, and then click **OK**.
3. Right-click the **M1** scoring task and select **Execute**.
4. When the information dialog box confirms that the report was created successfully, click **Close**. To view the results, click the **Results** tab and click **Result Set**.




If the scoring task was not successful, then review the **Log** tab for error messages.

5. Click the **Graphs** tab to graph the results.

   a. Click **Graph Wizard**, select **Histogram**, and then click **Next**.

   b. In the upper right corner, click **Use default assignments** and then click **Next**.

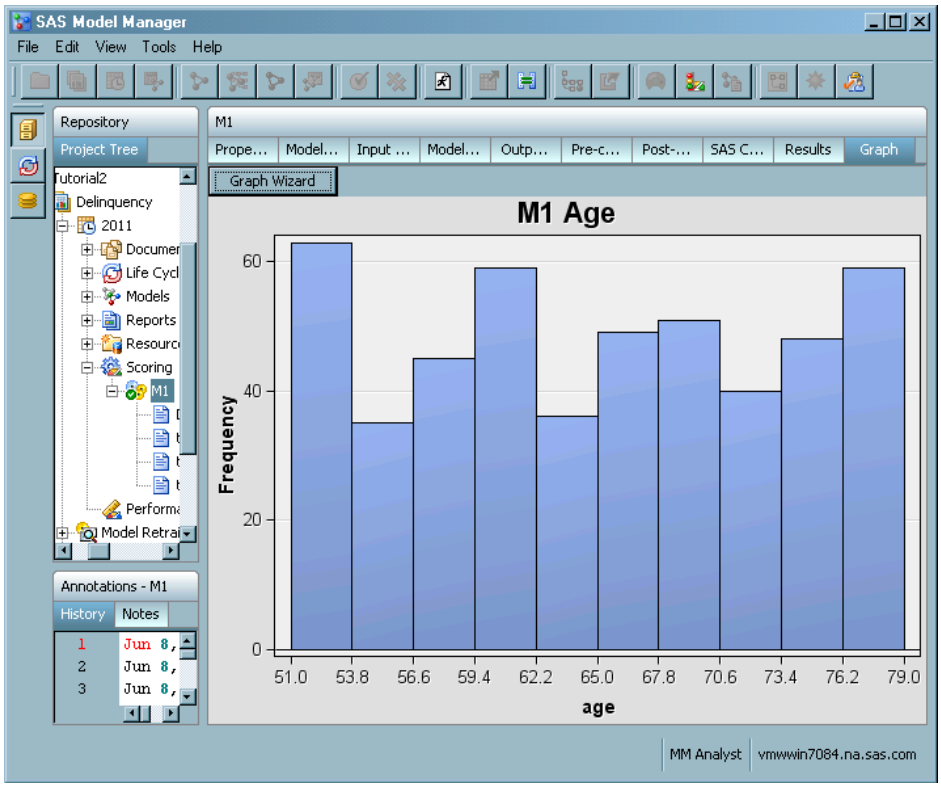
   c. Click the **Column name box** and select **age**.

   d. Click the **Operator** box and select **Greater than**.

   e. In the **Value** field, enter `50`. Click **Next**.

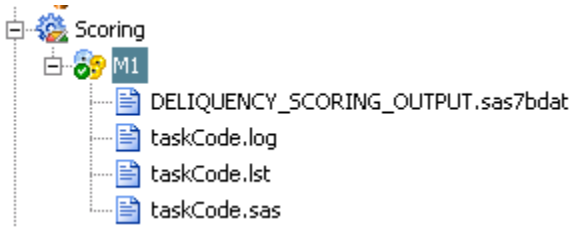
   f. In the **Title** field of the Chart Titles page, type `M1 Age`. Click **Next** and then click **Finish**.

      Here is the histogram on the **Results** tab:




6. Expand the **M1** scoring task to verify that four content files were saved and that the value for **Date Modified** is today's date.

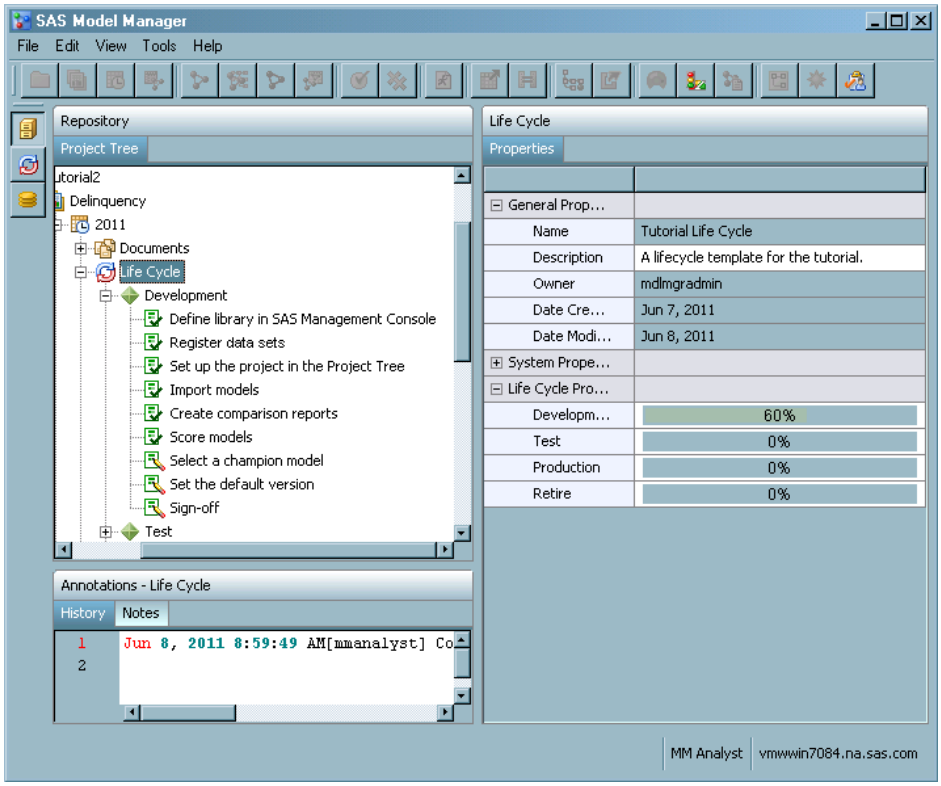
   Here is the **Scoring** folder and the files for the **M1** scoring task:




### *Update the Life Cycle (Optional)*

To update the Development milestone, follow these steps:

1. In the **Delinquency** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Score models** task. Click the **Status** box and select **Completed**.
3. Select **Score models**. The **Completed Date** and **Completed By** fields have been updated with today's date and your user ID.
4. Click the **Life Cycle** node to examine its properties. The value for **Modification Date** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.




### *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

# Declare a Champion Model

In this exercise you declare a champion model.

## *Set the Champion Model*

To set a champion model, follow these steps:

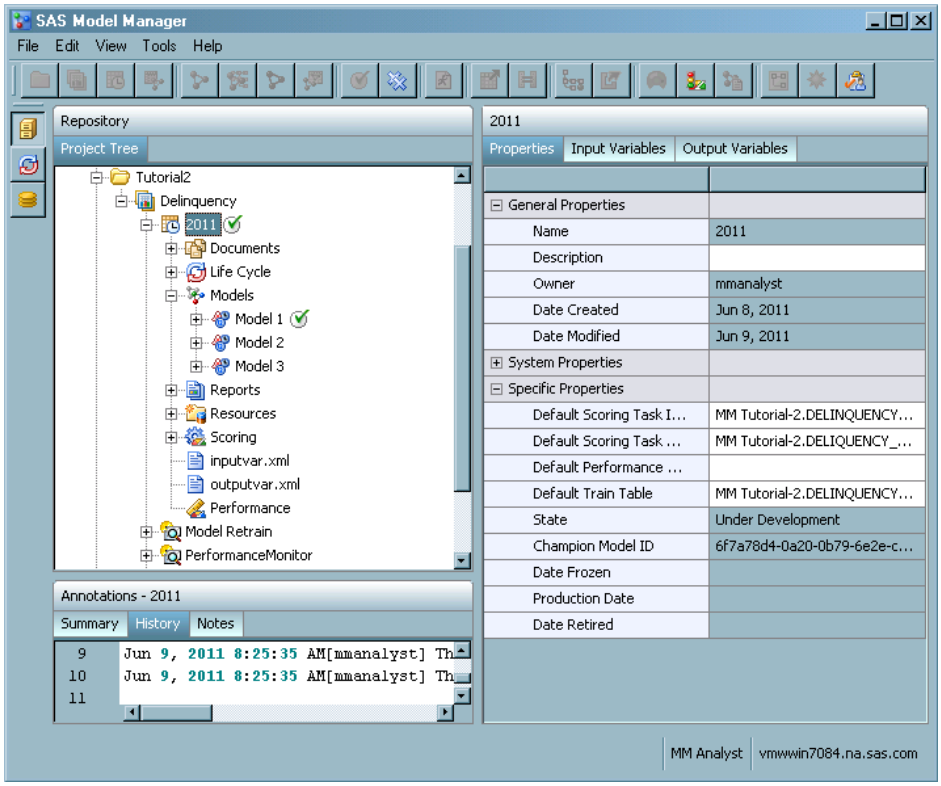
1. Expand the **Models** folder in the **2011** version. Right-click **Model 1**, select **Set Champion Model**, and click **Yes** to confirm.
2. Verify that the icon appears next to the champion model.
3. Select the version folder to examine its properties. The value for **Date Modified** is today's date. The value for the **Champion Model ID** is the champion model's UUID.

   TIP To document the reasons or assumptions for your selection of the champion model, use the version **Notes** tab. SAS Model Manager automatically annotates the History tab. For more information, see "Using the Annotation View" on page 55.

## *Set Default Version*

You assign a default version after the default champion model for the project is identified. To set the default version, follow these steps:

1. Right-click the **2011** version and select **Set Default Version**. Click **Yes** to confirm.
2. Verify that the icon appears next to the version folder.
3. Select the project folder to examine its properties. The value for **Date Modified** is today's date. The value for the **Default Version** is the name of the version folder.

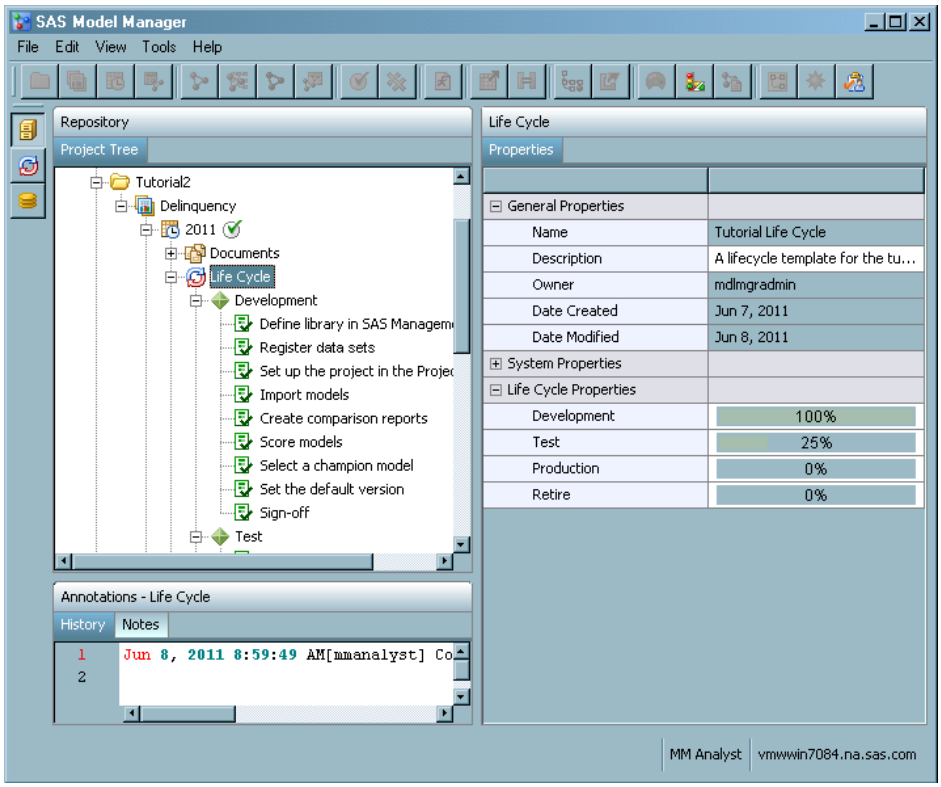



### *Update the Life Cycle (Optional)*

To update the Development milestone, follow these steps:

1. In the **Delinquency** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Select a champion model** task. Click the **Status** box and select **Completed**.
3. Select the **Set the default version** task. Click the **Status** box and select **Completed**.
4. Select the **Sign-off** task to indicate that all of the **Development** milestone tasks are complete. Click the **Status** box and select **Completed**.
5. Select the **Select a champion model**, **Set the default version**, and **Sign-off** tasks. The **Date Completed** and **Completed By** fields have been updated with today's date and your user ID.
6. Expand the **Test** milestone. Select the **Validate Score Input Data** task. Click the **Status** box and select **Completed**.

   *Note:* The **Select a champion model** task must have been completed before you can complete this task.
7. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** and **Test** properties display a bar chart that shows the percentage of completed tasks for this milestone.




## *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click  from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click  to claim an activity.

   *Note:* You can select an activity name and click  to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

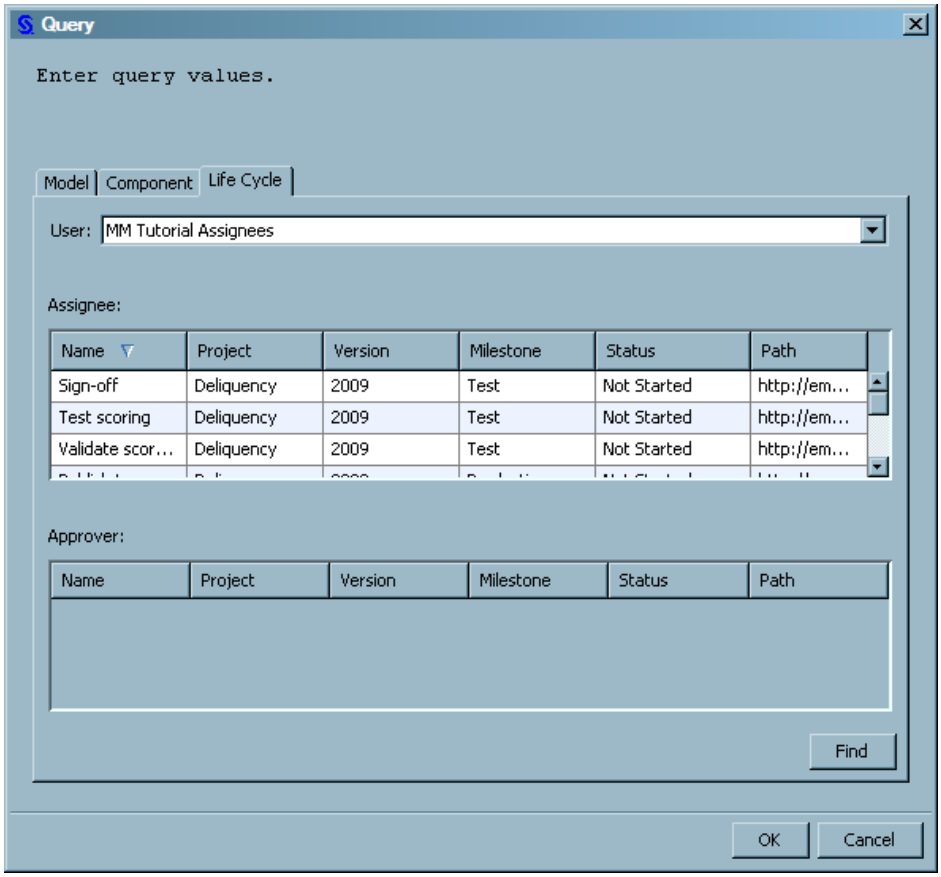
*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

## Query for the Remaining Project Tasks to Complete

In this example you search for the status of life cycle tasks by using the Query utility.

To search for the status of life cycle tasks, follow these steps:

1. Right-click the **Tutorial2** folder and select **Query**. The Query window appears.
2. Click the **Life Cycle** tab. Select the **User** box and select **MM Tutorial Assignees** and then click **Find**.
3. Examine the status of the associated milestones and click **OK**. The search results display tasks in the **Assignee** list that are assigned to the user and tasks in the **Approver** list that the user is assigned to approve. The **Assignee** query results return only the tasks that have a status of **Started** or **Not Started**. Results that have a status of **Complete** or **Approved** are omitted.



*Chapter 4*

# Tutorial 3: Importing and Exporting Models



## Overview of Importing and Exporting Models

SAS Model Manager provides several methods to import SAS models into a project version. You can import your models into a project version from the SAS Metadata Repository, SAS Enterprise Miner package files, SAS code and R models, and PMML

models. SAS macros are also available so that you can use SAS code to import or register SAS models into your project version.

After the champion model is validated, you can publish models to channels, export champion models to the SAS Metadata Repository, or publish a scoring function for a model to a database.

The tutorial provides examples and step-by-step directions for performing these tasks.

# Prerequisites

### *Tutorial 3 Models and Data Sets*

The exercises in this tutorial require that the Tutorial 3 data sets and models from SMM31Tutorial.zip be extracted and registered in SAS Management Console. If they have not been extracted and registered, see "Prepare Tutorial 3 Data Sets and Models" on page 8 to extract and register the files.

Importing models requires that you know where the SAS Model Manager administrator installed the Tutorial 3 models. If you do not know the location of the models, contact your SAS Model Manager administrator.

### *Verify Your User ID as a Member of SAS Model Manager User Groups*

This exercise ensures that your user ID is a member of the **MM Tutorial Assignees** group and the **Model Manager Advanced Users** group.

1. Open SAS Management Console and log on to the SAS Metadata Server.
2. On the **Plug-ins** tab, select **User Manager**.
3. In the right pane, double-click the **MM Tutorial Assignees** group and click the **Members** tab.
4. Review the **Current Members** list, and locate your user ID or a group that your user ID is a member of. If your user ID or group is not a member of the **MM Tutorial Assignees** group, ask your administrator to add you to this group. Close the properties window.
5. Find and double-click your user ID in the right pane of SAS Management Console.
6. Click the **Groups and Roles** tab. Review the **Member of** pane and locate the group **Model Manager Advanced Users**. If your user ID is not a member of this group, ask your administrator to add you to this group. Close the properties window.
7. Close SAS Management Console.

# Organize the Model Hierarchy

In this exercise, you create an organizational folder, a project, and a version for the modeling project.

### *Create a Folder*

To provide an organizational folder to manage your modeling projects, follow these steps:

1. Right-click the **MMRoot** node in the Project Tree and select **New** ⇨ **New Folder**. The New Folder dialog box appears.
2. Specify the following folder properties and click **OK**.

   **Name**
   enter `Tutorial3`.

   **Description**
   enter an optional folder description.

The new folder appears in the Project Tree.

### *Create a New Project*

To create a project, follow these steps:

1. Right-click the **Tutorial3** folder and select ⇨ **New** ⇨ **New Project**. The New Project Wizard appears.
2. Specify the following project properties and click **OK**:

   **Name**
   enter `Loan`.

   **Description**
   enter an optional description.

   **Model Function**
   select **Classification**. Click **Next**.
3. In Step 2 of the New Project Wizard, specify the project variables:
   a. Click the **Import Variables** button for the **Project Input Variables** table. Double-click **Shared Data** ⇨ **Model Manager** ⇨ **Tutorial3**. Select **HMEQ_PROJECT_INPUT** and click **OK**.
   b. Click the **Import Variables** button for the **Project Output Variables** table. Select **HMEQ_PROJECT_OUTPUT** and click **OK**.
   c. Click **Finish**.
4. Examine the **Tutorial3** folder to verify that it contains the **Loan** project.

### *Define the Project Properties*

To define the properties that SAS Model Manager uses to create reports and score models, follow these steps:

1. Select the **Loan** project in the **Tutorial3** folder and expand **Specific Properties** in the right pane.
2. Specify the default data tables and model variables for the project:

   **Default Test Table**
   select **HMEQ_TEST**.

**Default Train Table**
select **HMEQ_TRAIN**.

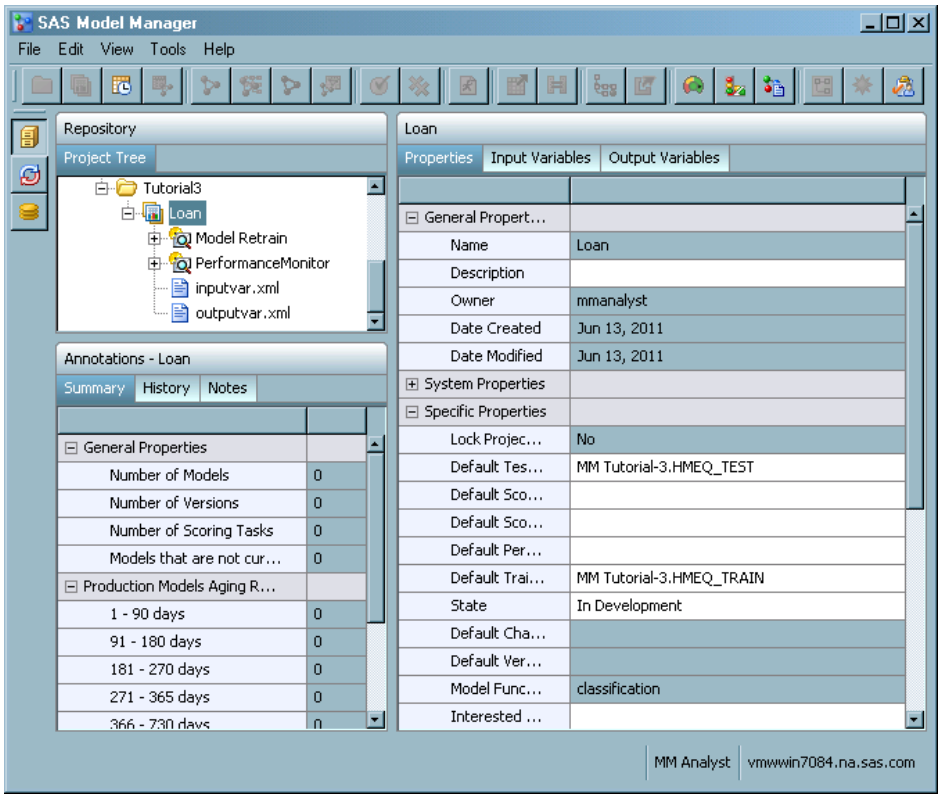
**Training Target Variable**
enter `bad`.

**Target Event Value**
enter 1.

**Class Target Level**
select **Binary**.

**Output Event Probability Variable**
select **score**.

Here are the project properties:




## *Create a Version*

Create a version for the project. The version folder contains life cycle information, auxiliary version documents, candidate model files, model comparison reports, resource files, scoring tasks, and model performance reports.

To create a new version, follow these steps:

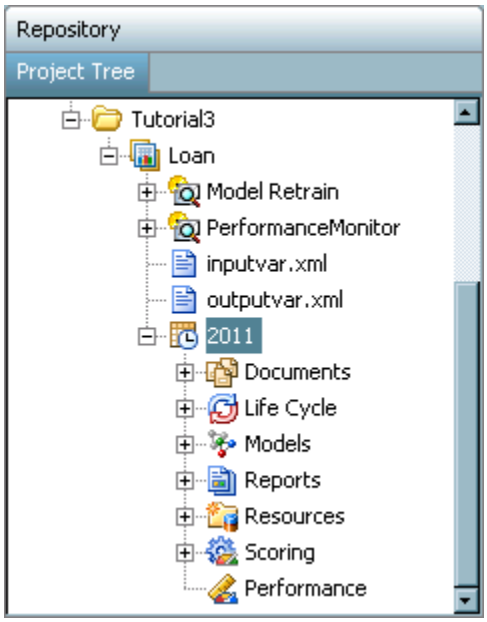
1. Right-click the **Loan** project and select **New** ⇨ **New Version**. The New Version dialog box appears.
2. Specify the following version properties and click **OK**.

   **Name**
   enter `2011`.

**Life Cycle Template**
select the user-defined template **Tutorial Life Cycle** that you created in the first tutorial. For more information, see "Create a Life Cycle Template" on page 23.

*Note:* If you are using a workflow process to track the progress of your project or version, you can select any life cycle template. You can then skip all tasks to update the life cycle.

3. Examine the **Loan** project to verify that it contains one version called **2011**. Select **Life Cycle**. Verify that the **Name** property is Tutorial Life Cycle.



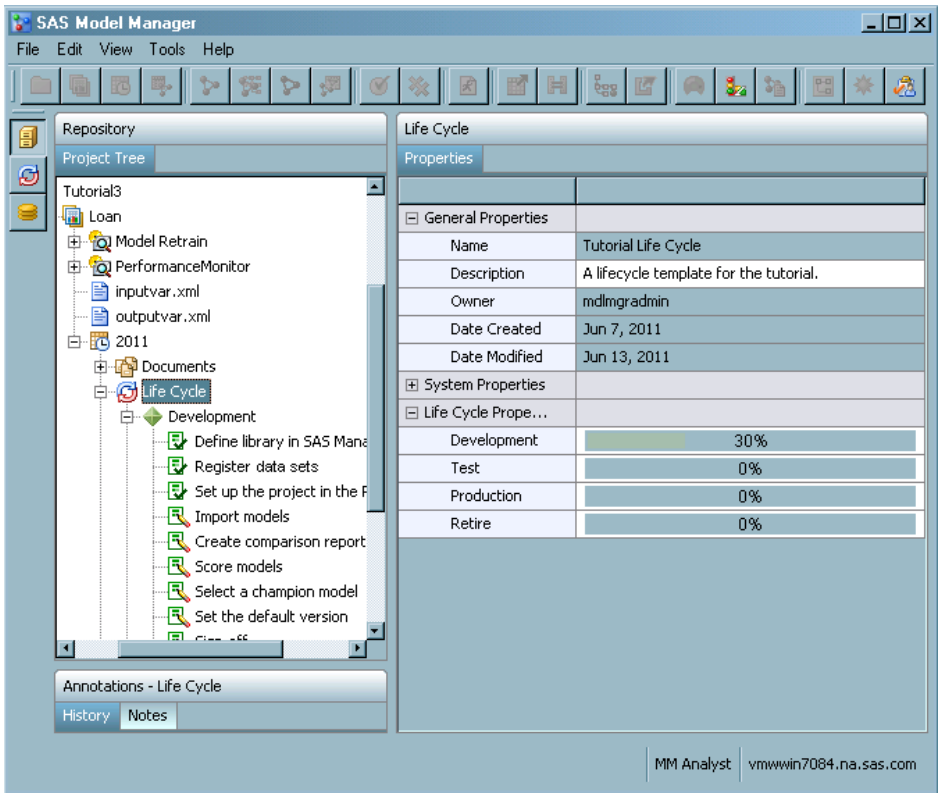

*Note:* If you want to use a workflow process to track the progress of your project or version, send a request to a SAS Model Manager administrator to create an instance of a workflow process definition to use for the tutorials. Include the name and UUID of the project or version with which you want to workflow process instance to be associated.

## *Update the Life Cycle (Optional)*

To complete the milestone task of adding data sources and setting up the project in the Project Tree, follow these steps:

1. In the **Loan** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Define library in SAS Management Console** task and examine the task properties. The **To Be Completed By** property, assigned in the life cycle template, determines which users or groups from the **Participants** list in the template editor are responsible for this milestone task. Because you are a member of the **MM Tutorial Assignees** group, you are authorized to update the task status.
3. Click the **Status** field and select **Completed**.
4. Select the **Register data sets** task and examine the task properties. Click the **Status** field and select **Completed**.
5. Select the **Set up project in the Project Tree** task. Click the **Status** box and select **Completed**.

6. Select all of the tasks whose status you updated and examine the properties. Verify that the value of the **Date Completed** property is today and that the value of the **Completed By** property is your user ID.
7. Select the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.



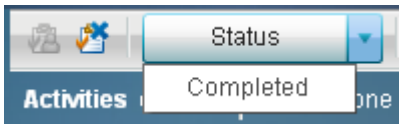

## *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.
2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.
3. (Optional) Enter a property value or change an existing property value in the Properties pane.
4. (Optional) Add a comment to the activity using the Comments pane.
5. Select a status value to complete the activity. The workflow process continues to the next activity.




6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

# Import Models

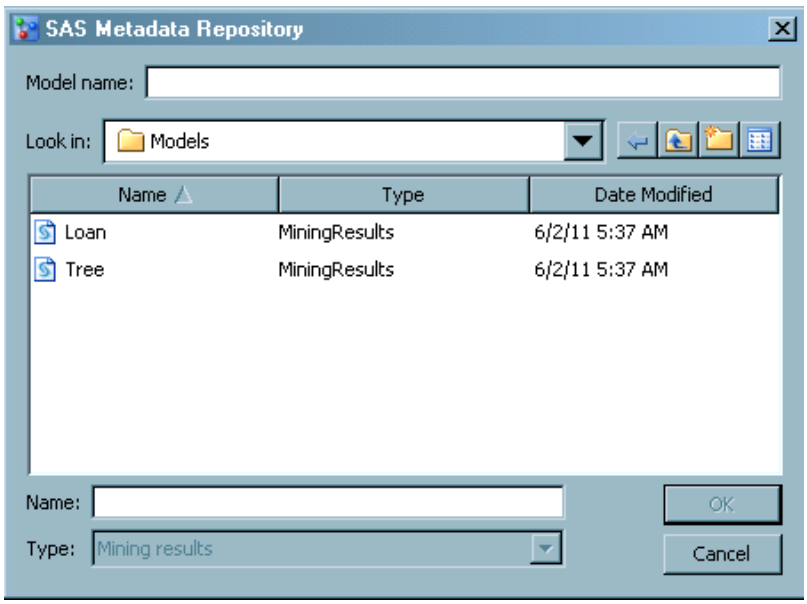
In this exercise you import models into SAS Model Manager from the SAS Metadata Repository, a PMML model file, and a SAS Enterprise Miner package file. Then you map the model variables. Before you import the model, verify that the model type is identical to the value of the project **Model Function** property, classification. For more information, see "Create a New Project" on page 71.

## *Import Models from a SAS Metadata Repository*

If your SAS Enterprise Miner 5.1 (or later) model files are registered in the SAS Metadata Repository, then you can use SAS Model Manager to import the files. If you do not have SAS Enterprise Miner models that have been registered in the SAS Metadata Repository, you can review these steps.

To import a model that is registered with SAS Enterprise Miner, follow these steps to understand the process:

1. Expand the **2011** version in the **Loan** project and right-click the **Models** folder. Then select **Import from ⇨ SAS Metadata Repository**. The SAS Metadata Repository dialog box appears.
2. Navigate to the location of the folder that contains the SAS Enterprise Miner models. Select a model from the folder.



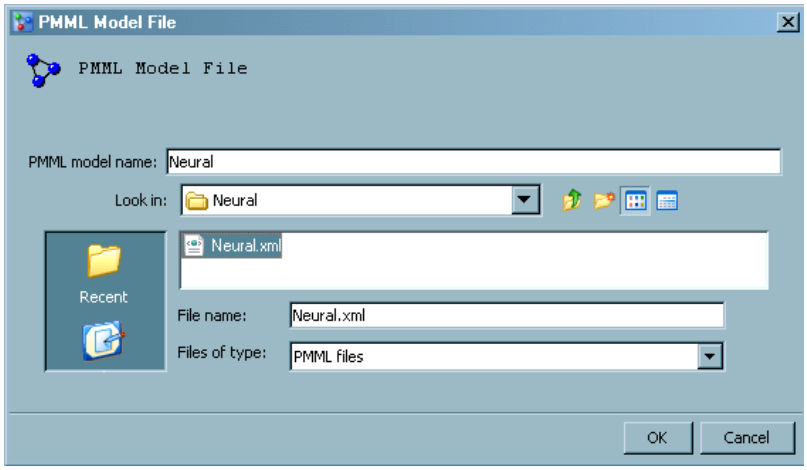
3. Click **OK**. After SAS Model Manager processes the request to import the model, the new model appears in the **Models** folder of your project's version.

## Import PMML Models

PMML (Predictive Modeling Markup Language) is an XML-based standard for representing data mining results. PMML is designed to enable the sharing and deployment of data mining results between vendor applications and across data management systems. You can use SAS Model Manager to import PMML 3.1 (or later) models that are produced by another software application, such as SAS Enterprise Miner.

To import a PMML model, follow these steps:

1. Expand the **2011** version in the **Loan** project and right-click the **Models** folder. Then select **Import From** ⇨ **PMML Model File**. The PMML Model File dialog box appears.
2. In the **PMML model name** field, enter **`Neural`**.
3. Navigate to the location of the folder that contains the PMML files. For this example, use **`<drive:>\Tutorial3\Samples\Neural\`** that was installed by the SAS Model Manager administrator. For more information, see "Prepare Tutorial 3 Data Sets and Models" on page 8.
4. Select the **`Neural.xml`** file and click **OK**.



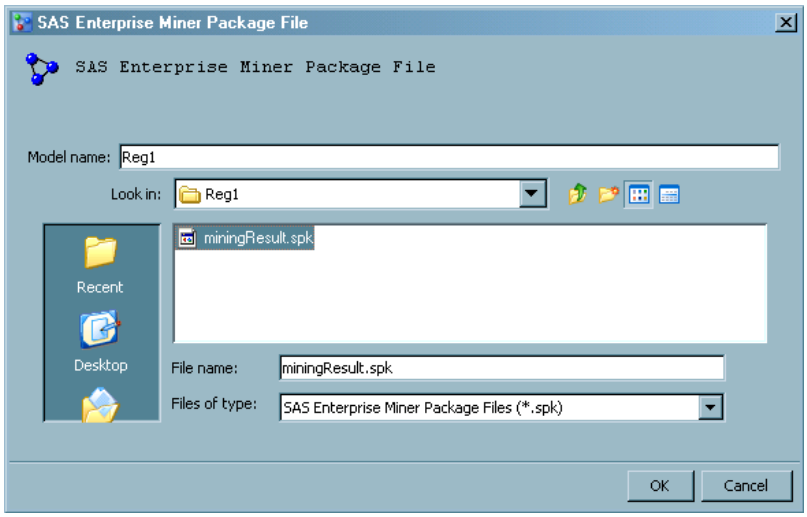

5. Examine the **Models** folder to verify that it contains the models. Right-click the folder and select **Expand All** to examine the model file.

## Import SAS Enterprise Miner Model Package Files

SAS Enterprise Miner package files, or SPK files, contain complete model information. You can import SAS Enterprise Miner models even if they are not registered in the SAS Metadata Repository. For information about how to create a package file, see the *SAS Model Manager: User's Guide*.

To import a SAS Enterprise Miner model that was saved as package file, follow these steps:

1. Expand the **2011** version in the **Loan** project and right-click the **Models** folder. Then select **Import From** ⇨ **SAS Enterprise Miner Package File**. The SAS Enterprise Miner Package File dialog box appears.
2. In the **Model Name** field, enter **`Reg1`**.
3. Navigate to the location of the folder that contains the SAS Enterprise Miner package files. For this example, use **`<drive:>\Tutorial3\Samples\Reg1`** that was installed by the SAS Model Manager administrator. For more information, see "Prepare Tutorial 3 Data Sets and Models" on page 8.
4. Select the **`miningResult.spk`** file and click **OK**.




5. Repeat steps 2 through 4 to import a second package file that is located in **`<drive:>\Tutorial3\Samples\Tree1`**. Name the model **Tree 1**.
6. Examine the **Models** folder to verify that it contains the models. Right-click the folder and select **Expand All** to examine the model files.

## *Map Model Variables to Project Variables*

When the names for the model output variable are not identical to the names for the project output variables, you must map the variables.

To map model output variables to project output variables, follow these steps:

1. Map model variables for the first model. Select **Reg1** in the **Models** folder, click the **Model Mapping** tab in the right pane, and click **Edit**. Set the following mapping and click **OK**:

| Project Variables | Model Variables |
|---|---|
| **score** | **EM_EVENTPROBABILITY** |

2. Map model variables for the second model. Right-click **Tree 1** in the **Models** folder, and select **Set Model Output Mapping**. Set the following mapping and click **OK**:

| Project Variables | Model Variables |
|---|---|
| score | EM_EVENTPROBABILITY |

### Update the Model Life Cycle (Optional)

To complete the milestone task for adding the models, follow these steps:

1. In the **Loan** project, expand **2011** ⇨ **Life Cycle** ⇨ **Develop**.
2. Select the **Import models** task. Select the **Status** box and select **Completed**. The **Date Completed** and **Completed By** fields have been updated with today's date and your user ID.
3. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.

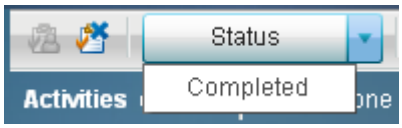
### Update the Workflow Process (Optional)

To complete the activities in the associated workflow process, follow these steps:

1. Click  from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.
2. From the Activities category view, select an activity name, and click  to claim an activity.

   *Note:* You can select an activity name and click  to release an activity that you had previously claimed.
3. (Optional) Enter a property value or change an existing property value in the Properties pane.
4. (Optional) Add a comment to the activity using the Comments pane.
5. Select a status value to complete the activity. The workflow process continues to the next activity.




6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

# Declare a Champion Model

In this exercise, you declare a champion model in the model and version folders.

## *Set the Champion Model*

To assign a champion model, follow these steps:

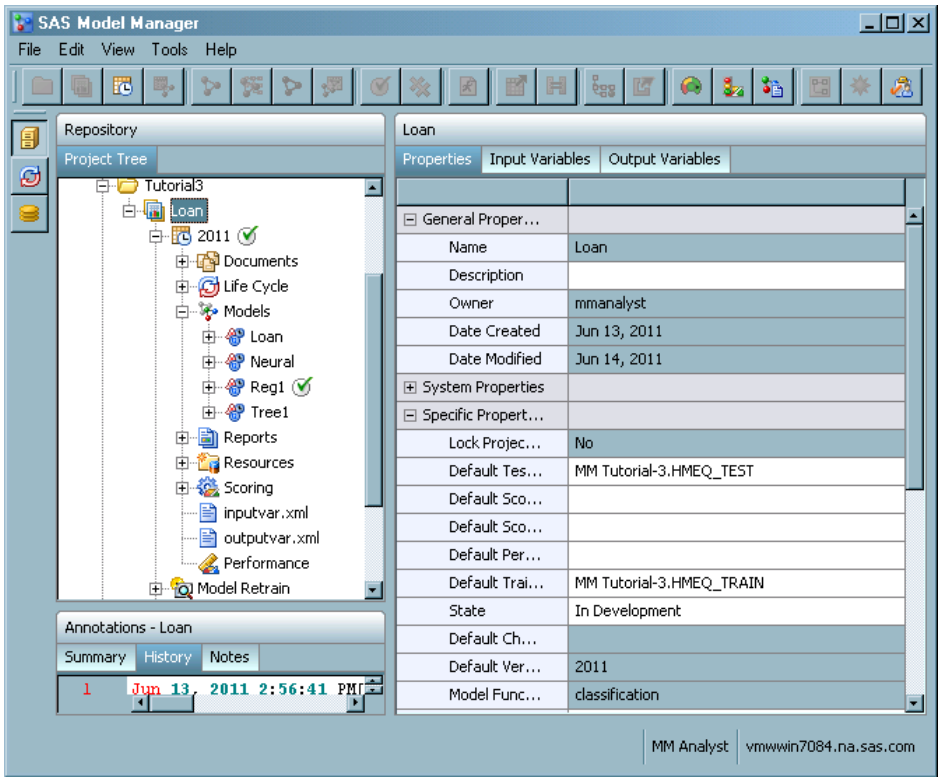
1. Expand the **Models** folder in the **2011** version of the **Loan** project. Right-click **Reg1**, select **Set Champion Model**, and click **Yes** to confirm.
2. Verify that the ✓ icon appears next to the champion model.

## *Set Default Version*

You assign a default version after the default champion model for the project is identified. To set the default version, follow these steps:

1. Right-click the **2011** version and select **Set Default Version**. Click **Yes** to confirm.
2. Verify that the ✓ icon appears next to the version folder.
3. Select the project folder to examine its properties. The value for **Date Modified** is today's date. The value for the **Default Version** is the name of the version folder.

   *Note:* The champion model for the default version is used as the champion model for the project.



**TIP** SAS Model Manager automatically annotates the **History** tab. To document the reasons or assumptions for your selection of the default version, use the project **Notes** tab. For more information, see "Using the Annotation View" on page 55.

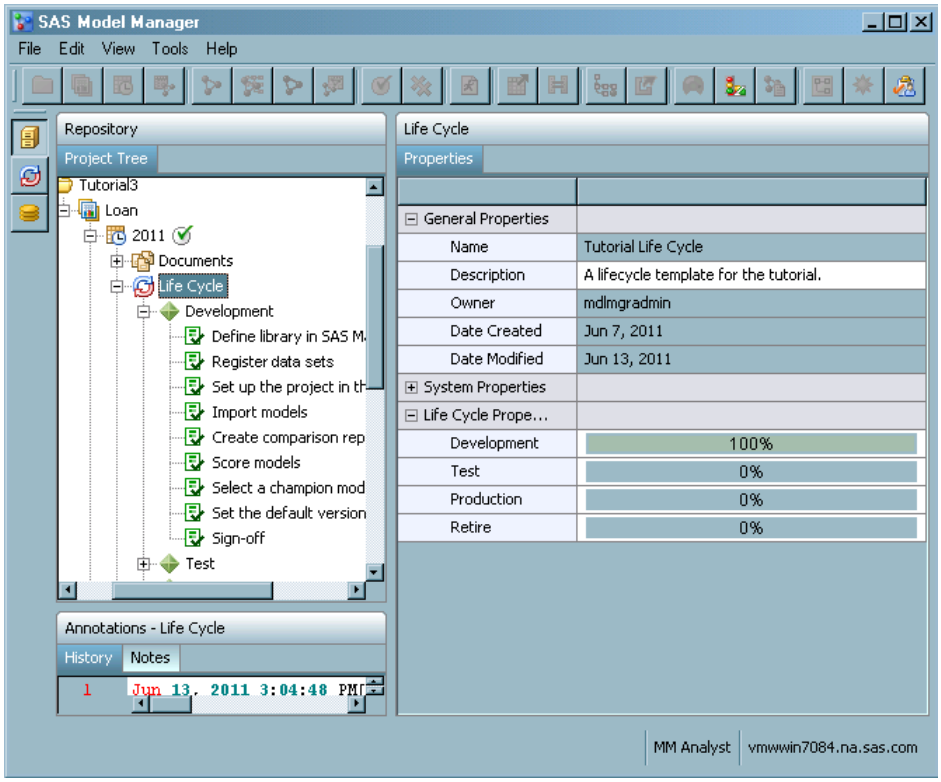
### *Update the Life Cycle (Optional)*

To update the life cycle milestones, follow these steps:

1. In the **Loan** project, expand **2011** ⇨ **Life Cycle** ⇨ **Development**.
2. Select the **Create comparison reports** task. Click the **Status** box and select **Completed**.

   *Note:* Although this task was not part of this exercise, dependencies in the life cycle require that you mark this task complete.
3. Select the **Score models** task. Click the **Status** box and select **Completed**.

   *Note:* Although this task was not part of this exercise, dependencies in the life cycle require that you mark this task complete.
4. Select the **Select a champion model** task. Click the **Status** box and select **Completed**.
5. Select the **Set the default version** task. Click the **Status** box and select **Completed**.
6. Select the **Sign-off** task. Click the **Status** box and select **Completed**.
7. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Development** property displays a bar chart that shows the percentage of completed tasks for this milestone.



### Update the Workflow Process (Optional)

To complete the activities in the associated workflow process, follow these steps:

1. Click from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

   *Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

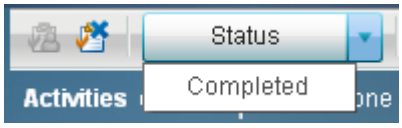
2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.

4. (Optional) Add a comment to the activity using the Comments pane.

5. Select a status value to complete the activity. The workflow process continues to the next activity.




6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

## Deliver Models

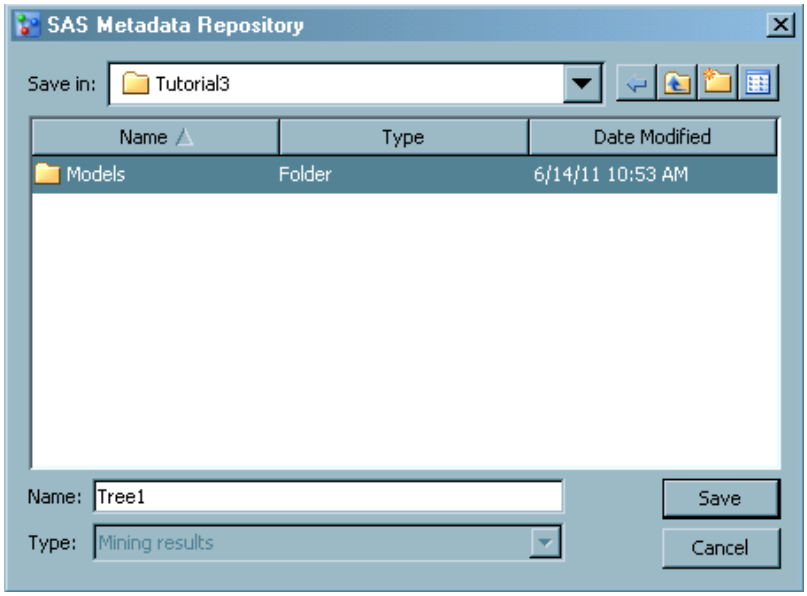
In this exercise, you use the comprehensive publishing environment for model delivery to share models. Model delivery most often includes model score code and its associated input and output metadata. You export a model and the champion model for a project to the SAS Metadata Repository and publish a model to a publish channel. For each model that you export, SAS Model Manager creates a MiningResult object in the SAS Metadata Repository. Only models that have a **Score Code Type** of **Data Step** can be exported to the metadata repository. Application software, such as SAS Data Integration Studio or SAS Enterprise Guide, can access the MiningResult object through the SAS Metadata Server and submit on-demand or batch scoring jobs.

### Export a Model to the SAS Metadata Repository

SAS Model Manager uses the SAS Folder view to export the model to any folder that is accessible to the user. You can export a model to folders in the SAS Foundation repository or to folders in custom repositories that are created in SAS Management Console to reflect the structure of your business organization.

To export a model to a SAS Metadata Repository, follow these steps:

1. Expand **Loan** ⇨ **2011** ⇨ **Models** and right-click the **Tree 1** model. Then select **Export Model**. The SAS Metadata Repository dialog box appears
2. Navigate to the folder where you want to store the model.




3. Enter **Tree 1** as the name and click **Save**. If a MiningResult object is in the repository that has the same name, then you are prompted to decide whether to overwrite the metadata for this stored object.

   ***CAUTION:***
   **Do not overwrite an existing MiningResult object unless you are certain that the model is from the same project in SAS Model Manager.**

4. In theExport Model message box, click **Close**.

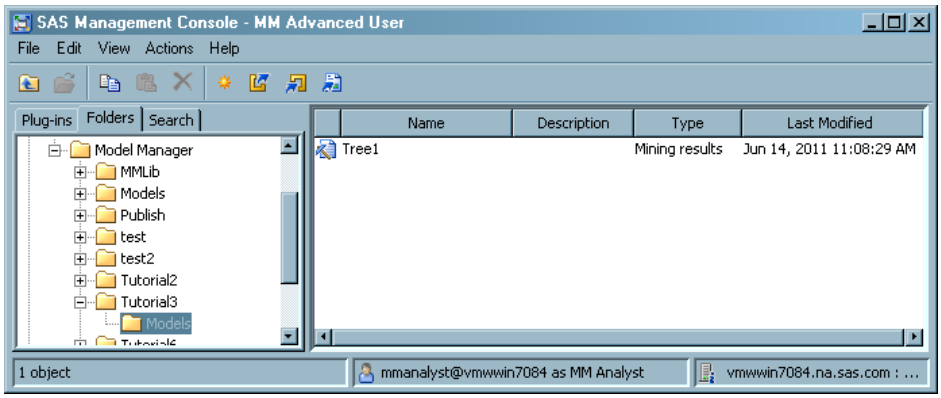
If you change the score code for the model, then export the model again to ensure that your score application uses the current scoring code.

## *Verify the Model Export*

To verify that SAS Model Manager successfully created the MiningResult object in the metadata repository for an exported model, use SAS Management Console. To view the contents of the exported model or project, you can use SAS Data Integration Studio. You can also use SAS Management Console to export the MiningResult object to a SAS package.

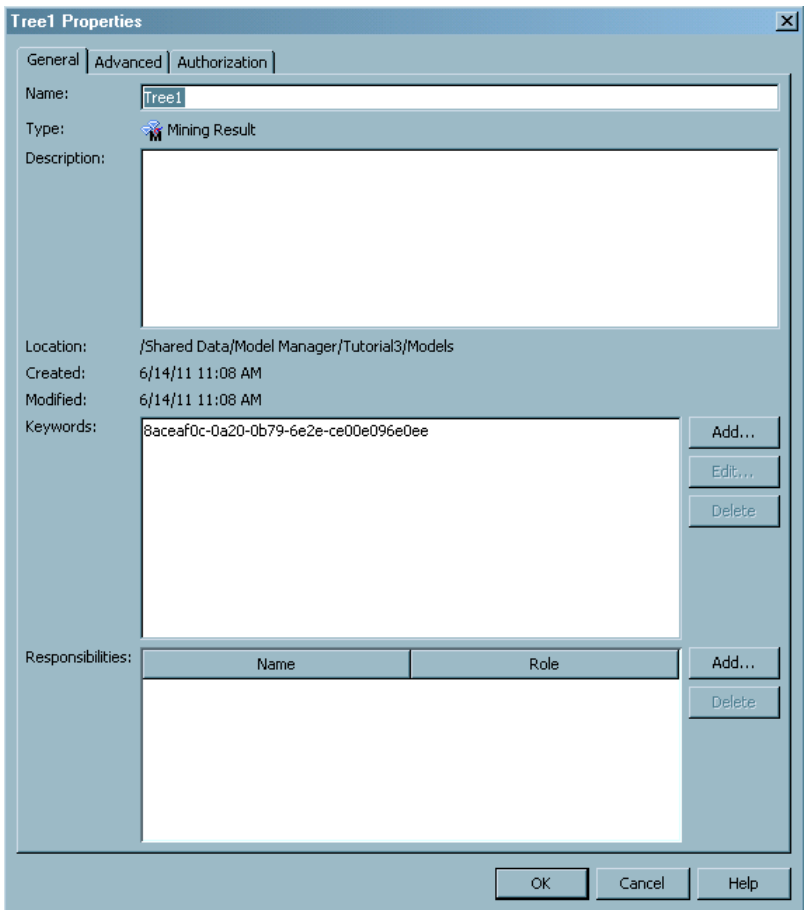
To view a MiningResult object in the metadata repository, follow these steps:

1. Open SAS Management Console and log on to the SAS Metadata Server using the same user ID that you use to log on to SAS Model Manager.
2. Click the **Folders** tab and expand the folders to locate the model that you exported. When you select the folder, the right pane lists the MiningResult objects for the exported models.




3. Right-click the **Tree 1** MiningResult object and select **Properties** from the pop-up menu. The Properties window appears.

4. Examine the **Keywords** box on the **General** tab to verify that the MiningResult object contains the universal unique identifier (UUID) of the exported model. The UUID is a system property that SAS Model Manager automatically assigns to each model. To view a system property in SAS Model Manager, click the + icon beside the **System Properties** heading to expand the section.




**TIP** You can use the UUID to conduct filtered searches and query the exported models. For more information, see the *SAS Model Manager: User's Guide*.

5. Examine the metadata on the **Advanced** tab to determine when the MiningResult object was created or most recently updated.

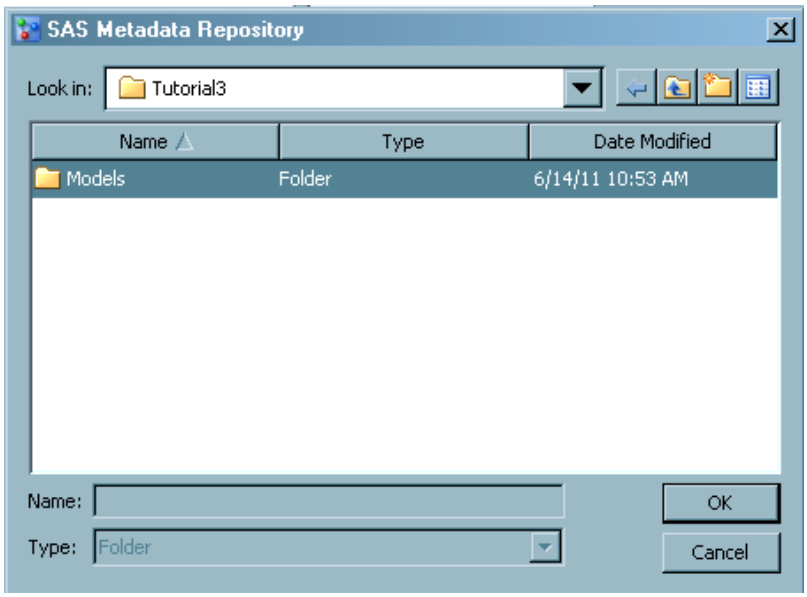
6. Click **OK**.

### Export the Champion Model

To export the champion model for a project, you must have already assigned the default version for the project. SAS Model Manager examines the project and always exports the champion model in the default version. When the default version for a project changes and you export the model again at the project level, the scoring application automatically uses the latest score code. SAS Model Manager exports only models whose model property **Score Code Type** is set to **Data Step**.

To export the champion model for a project, follow these steps:

1. Verify that the project has a default version assigned. Select the **Loan** project folder to examine its properties. The **Default Version** property contains the name of the default version.
2. Right-click the **Loan** project and select **Export Project Champion Model** from the pop-up menu. Click **Yes** for the information message that the project is unlocked. The SAS Metadata Repository dialog box appears.
3. Navigate to the folder where you want to store the model.




4. Select the folder and click **OK**. If a MiningResult object is in the repository that has the same name, then you are prompted to decide whether to overwrite the metadata for this stored object.

   ***CAUTION:***
   **Do not overwrite an existing MiningResult object unless you are certain that the model is from the same project in SAS Model Manager.**

5. In the information message box, click **Close**.

### Publish Models

SAS Model Manager uses SAS Publishing Framework to publish models to defined channels. SAS Model Manager creates a SAS Package file (SPK) for the model in a publication channel. Authorized users who subscribe to the channel can choose to receive e-mail notifications when updated models are ready to deploy to testing or

production scoring servers and when the SPK file is published to a publication channel. Then you can extract and validate the scoring logic, deploy champion models to a production environment, and monitor the performance of your models.

To publish a model to a channel, follow these steps:

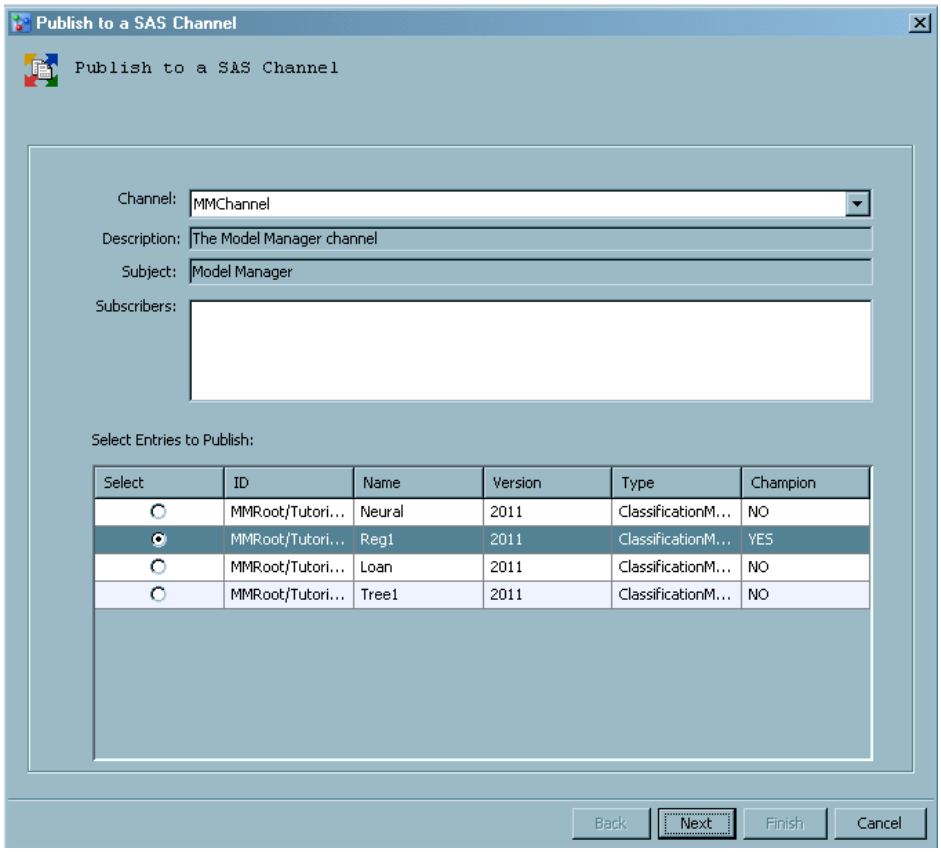
1. Expand **Loan** ⇨ **2011** and right-click the **Models** folder. Then select **Publish**. The Publish to a SAS Channel window appears.

   TIP You can publish models from the organization, project, version, or model folder in the Project Tree.

2. Select a publication channel from the **Channel** list.

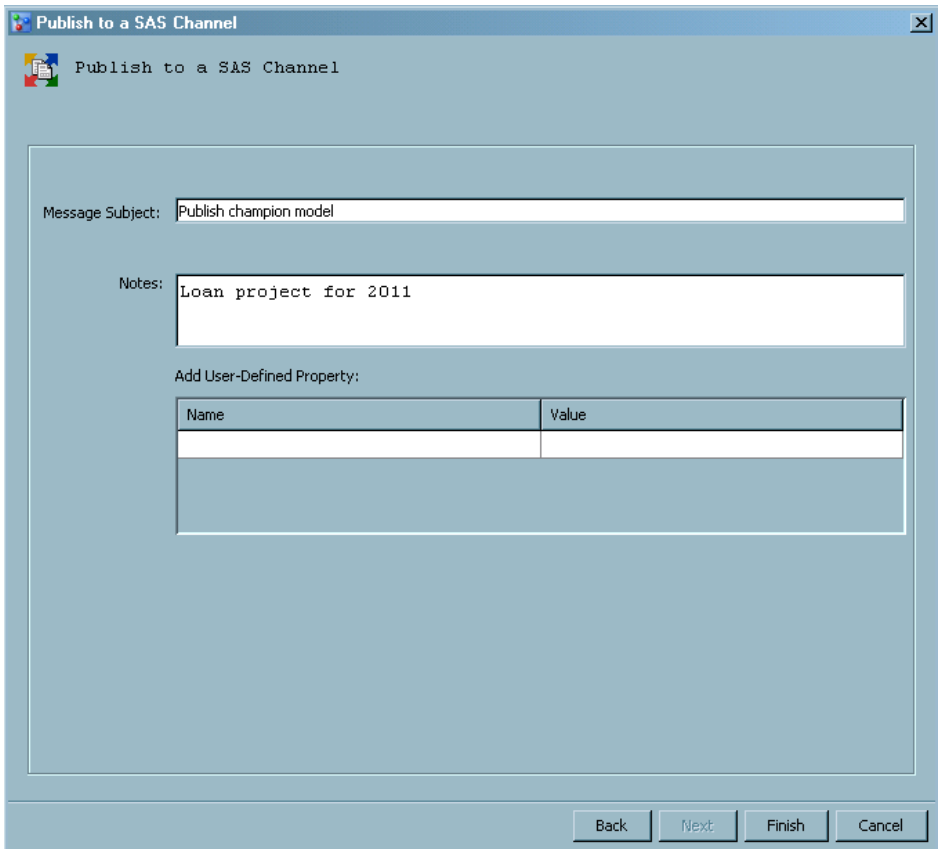
   *Note:* The channel values for **Description**, **Subject**, and **Subscribers** are defined in the SAS Metadata Repository with SAS Management Console.

3. Select **Reg1** as the model to publish in the **Select Entries to Publish** table. SAS Model Manager lists all of the models in the version folder. To view the entire folder name, expand the ID column heading. Click **Next**.

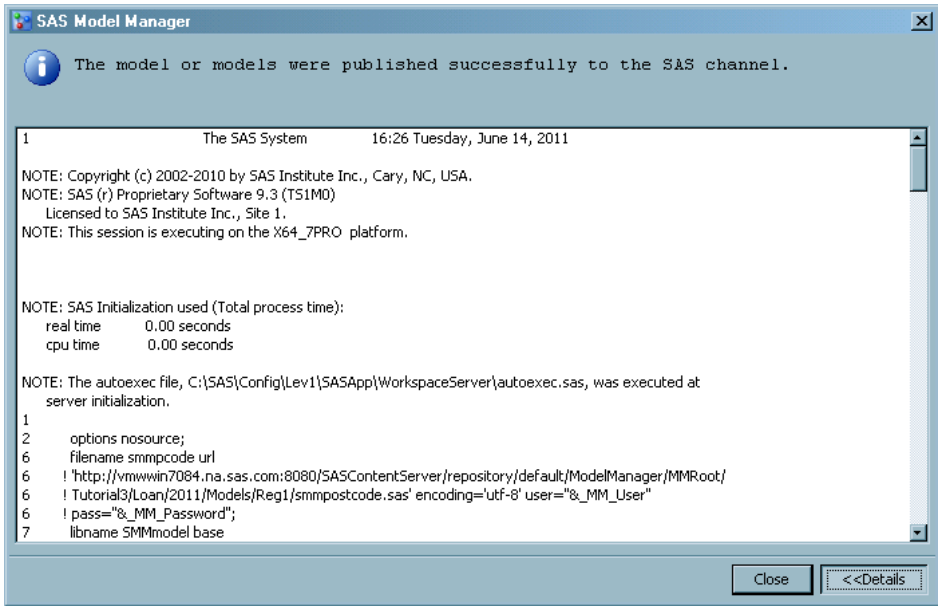



4. Specify an optional subject line for the e-mail message in the **Message Subject** box. For this example, enter `Publish champion model`. By default, SAS Model Manager uses the value that is defined in the publication channel. If you omit the subject line, then the name of the published model is used.

5. In the **Notes** box include information about the model that might be useful to other users who are involved with the project. For this example, enter `Loan project for 2011`.




6. Click **Finish**. The information dialog box appears and provides information about whether SAS Model Manager successfully published the model. Click **Details** to display a log of the publication process and any messages.




7. Click **Close**.

The SAS package that is sent to the publication channel contains the model input, output, SAS code, and its properties. You can submit a SAS DATA step program that calls the SAS Publish API (Application Programming Interface) to extract and deploy the model to a testing or scoring server. SAS Model Manager also provides a SAS macro

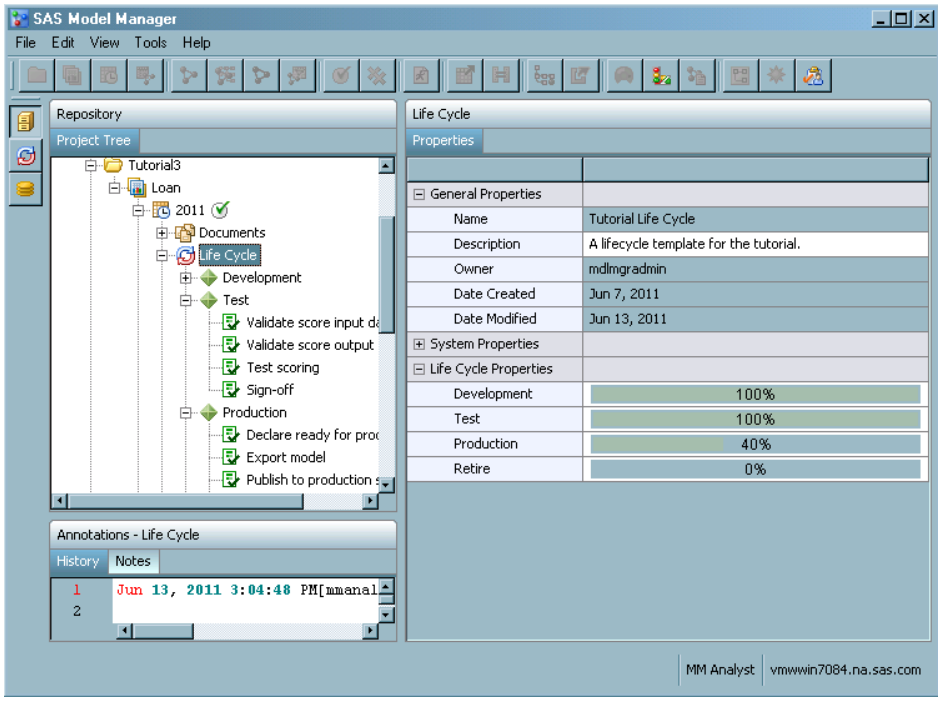
program, called MM_GetModels, that extracts the SAS code and metadata to score the model. Typically, extracted files are placed on a local drive of the scoring server that is used to deploy the published model. For more information, see the *SAS Model Manager: User's Guide*.

### *Update the Life Cycle (Optional)*

To update the life cycle milestones, follow these steps:

1. In the **Loan** project, expand **2011** ⇨ **Life Cycle** ⇨ **Test**.
2. Select each milestone task for **Test**. Click the **Status** box and select **Completed**.

   *Note:* Although this task was not part of this exercise, dependencies in the life cycle require you to mark this task complete.
3. Expand the **Production** milestone. Select the **Declare ready for production** task. Click the **Status** box and select **Completed**.
4. Select the **Export model** task. Click the **Status** box and select **Completed**.
5. Select the **Publish to production score server** task. Click the **Status** box and select **Completed**.
6. Click the **Life Cycle** node to examine its properties. The value for **Date Modified** is today's date. The **Test** and **Production** properties display bar charts that show the percentage of completed tasks for these milestones.




### *Update the Workflow Process (Optional)*

To complete the activities in the associated workflow process, follow these steps:

1. Click  from the SAS Model Manager main window to view the workflow process activities in your workflow inbox. Workflow Console is launched in a Web browser, and displays the Activities category view.

*Note:* The list displays only the activities for which you are the actual owner or are assigned as a potential owner, and that have the state of **`Started`**.

2. From the Activities category view, select an activity name, and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed.

3. (Optional) Enter a property value or change an existing property value in the Properties pane.
4. (Optional) Add a comment to the activity using the Comments pane.
5. Select a status value to complete the activity. The workflow process continues to the next activity.


6. Repeat steps 2 through 5 for the activities that you completed during this tutorial.

*Note:* For more information, see Chapter 11, "Tutorial 10: Using Workflow Console," on page 165.

*Chapter 5*

# Tutorial 4: Using Advanced Reporting



## Overview of Advanced Reporting

The advanced reporting capability of SAS Model Manager enables you to create two different types of reports.

- User-defined reports enable your company to add enterprise-specific reports to the existing list of reports that are available via the New Report Wizard dialog box.
- Ad hoc reports enable you to create one-of-a-kind reports as you need them.

To make it easy to create these reports, SAS Model Manager provides a number of SAS macro variables and SAS macro programs. These macros can be used to gain access to model-specific information as well as to more general folder and user information.

This tutorial shows you the basic tasks that are involved in creating new SAS Model Manager reports. It contains examples and step-by-step directions about setting up and running ad hoc and user-defined reports.

## Prerequisites

### *Models Used in Tutorial 2*

The exercises in this tutorial depend on some of the properties of the specific models that were added in Tutorial 2. Use the projects, versions, or models that are specified here.

This tutorial is designed to follow Chapter 3, "Tutorial 2: Performing Basic SAS Model Manager Tasks," on page 37.

### The Required Tutorial Files

The SAS programs that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

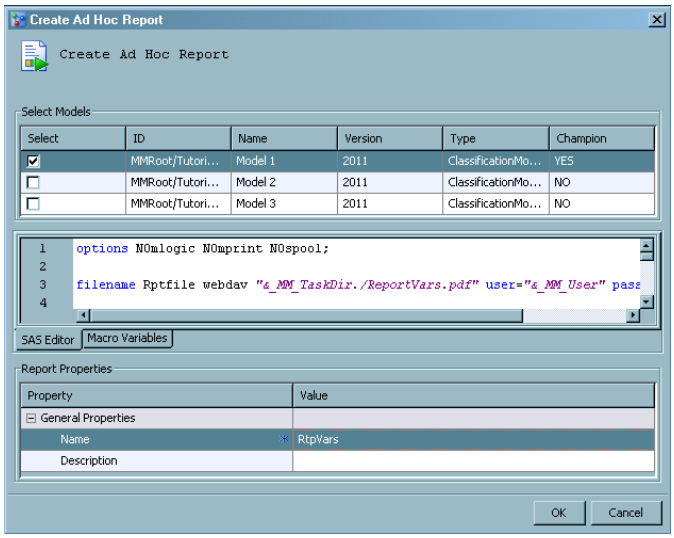
This tutorial requires the following files in the **`<drive>\Tutorial4\Samples`** folder:

- RptVars.sas
- ScoreRange.sas
- ScoreRangeMacro.sas
- ScoreRangeTemplate.xml

## Create a Simple Ad Hoc Report

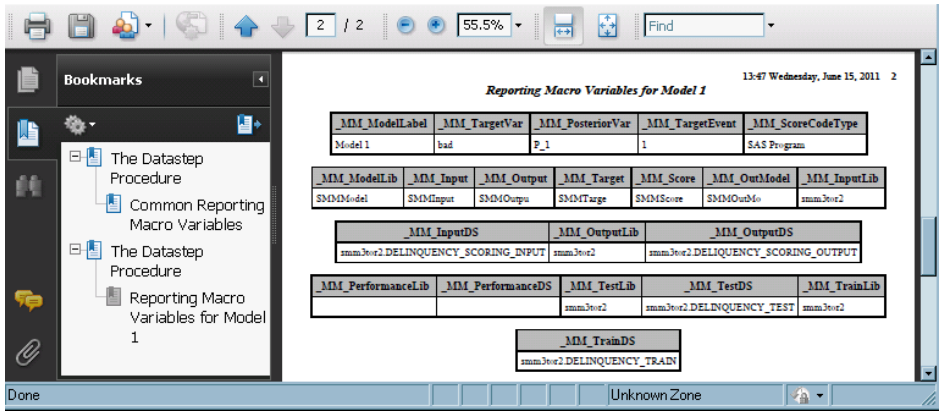
In this exercise, you create a PDF file to display the macro variables that are available in the SAS Model Manager reporting environment:

1. In the **`<drive>/Tutorial4/Samples`** folder, open the tutorial example report RptVars.sas in a text editor.
2. Copy the code from RptVars.sas.
3. Log on to SAS Model Manager.
4. Open the Create Ad Hoc Report window:
   a. Expand the **Tutorial2** folder, the **Delinquency** project, and the **2011** version.
   b. Right-click the **Reports** folder and select **Reports** ⇨ **Create Ad Hoc Report**.
5. In the Create Ad Hoc Report window, select **Model 1** in the **Select Models** table.
6. In the **SAS Editor**, paste the code that you copied in Step 2.
7. In the **Name** field of the **Report Properties** table, enter **`RptVars`**.




8. Click **OK**. SAS Model Manager creates the report and highlights the report in the **Reports** folder. Click **Close** in the information message.
9. To view the report, expand the new report **RptVars**, right-click **RptVars.pdf,** and select **Open**.

Here is page 2 of the PDF report output:




## Create an Ad Hoc Score Range Report

In this exercise, you create an ad hoc report to categorically display score ranges in an HTML report. To create output in HTML from an ad hoc report, ensure that the report code is enclosed by a SAS Model Manager formatting macro. To do this, follow these steps:

1. In the **`<drive>/Tutorial4/Samples`** folder, open the example report ScoreRange.sas and copy the code.
2. If necessary, log on to SAS Model Manager.
3. In the Project Tree, expand the **Tutorial2** folder, the **Delinquency** project, and the **2011** version.
4. Right-click the **Reports** folder and select **Reports** ⇨ **Create Ad Hoc Report**.

5. In the Create Ad Hoc Report window, select **Model 1** in the **Select Models** table.
6. In the **SAS Editor**, paste the code that you copied in Step 1.
7. Modify the Score Range code to format the report in HTML.

   The ScoreRange.sas program uses the SAS Model Manager formatting macros, which enable user reports to be formatted in PDF, HTML, RTF, and Excel. A beginning formatting macro code precedes the report code. The ending formatting macro must be the last line of code in the report program.

   a. Add the argument **`reportFormat=html`** to the %MM_ExportReportsBegin macro argument list. Here is the modified macro:

   ```
   %MM_ExportReportsBegin(reportFormat=html, fileName=ScoreRange);
   ```

   b. Add the argument **`reportFormat=html`** to the %MM_ExportReportsEnd macro argument list. Here is the modified macro:

   ```
   %MM_ExportReportsEnd(reportFormat=html);
   ```

8. In the **Name** field of the **Report Properties** table, enter **`ScoreRange`**.
9. Click **OK**. SAS Model Manager runs the report and creates the **ScoreRange** folder under the **Reports** folder.
10. To view the **ScoreRange** report, expand the **ScoreRange** folder, right-click **ScoreRange.html**, and select **Open**.

    Here is the output from the FREQ procedure as a table and as a graph:

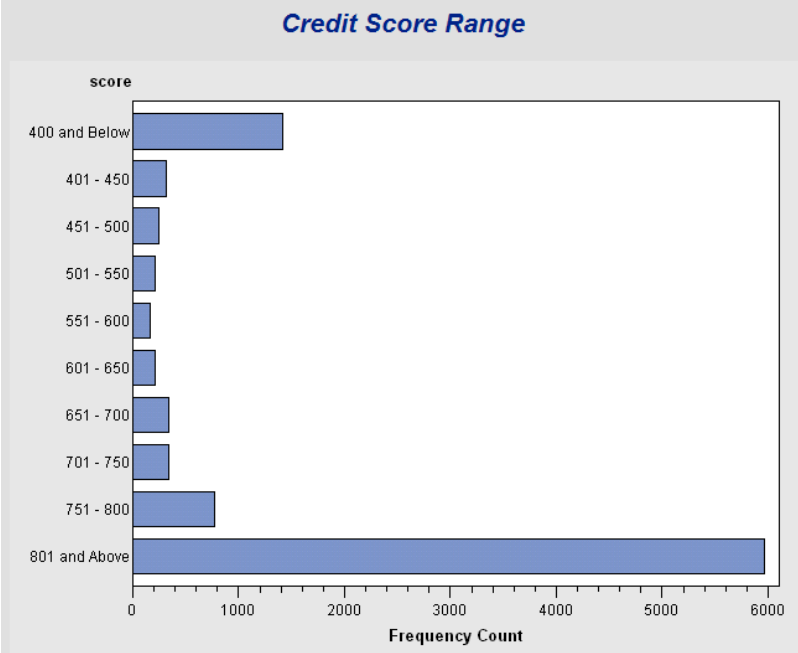
***Display 5.1*** *The Score Range Report Table*

**Credit Score Range**

**The FREQ Procedure**

| score | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
|---|---|---|---|---|
| 400 and Below | 1419 | 14.19 | 1419 | 14.19 |
| 401 - 450 | 322 | 3.22 | 1741 | 17.41 |
| 451 - 500 | 249 | 2.49 | 1990 | 19.90 |
| 501 - 550 | 206 | 2.06 | 2196 | 21.96 |
| 551 - 600 | 161 | 1.61 | 2357 | 23.57 |
| 601 - 650 | 213 | 2.13 | 2570 | 25.70 |
| 651 - 700 | 340 | 3.40 | 2910 | 29.10 |
| 701 - 750 | 344 | 3.44 | 3254 | 32.54 |
| 751 - 800 | 777 | 7.77 | 4031 | 40.31 |
| 801 and Above | 5969 | 59.69 | 10000 | 100.00 |

***Display 5.2*** *The Score Range Report Graph*



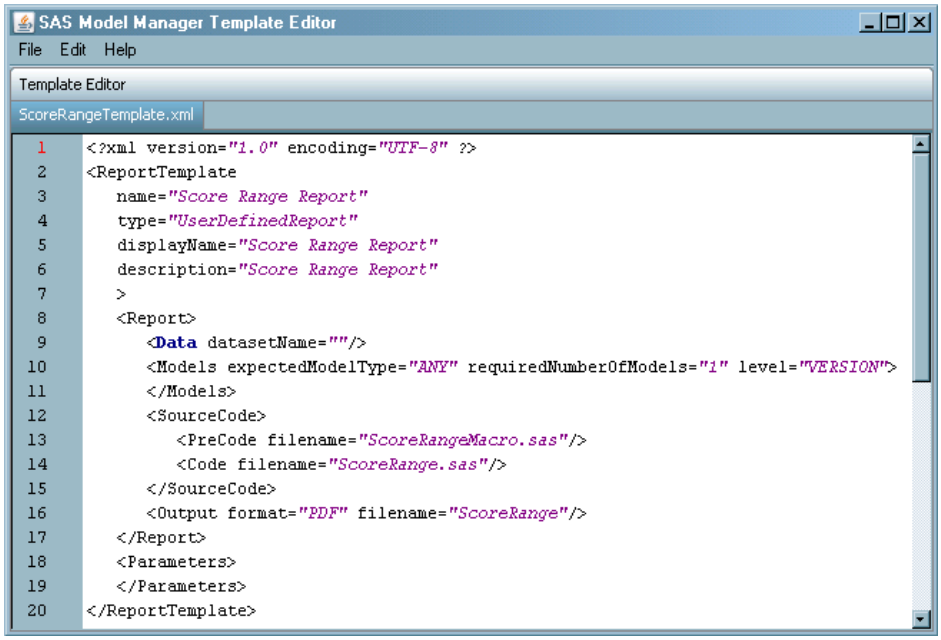

# Install a User-defined Score Range Report

In this exercise, you upload the Score Range report to SAS Content Server. After the template is uploaded, you can run the Score Range report from the New Reports Wizard. This exercise has two parts. In the first exercise, you upload the report files to the SAS Content Server. In the second exercise, you create a Score Range report from the New Reports Wizard. To upload a report XML file or SAS file, you must have a user ID that is in the Model Manager Administrator Users group.

## *Install a User-defined Report*

To upload the Score Range report to the New Reports Wizard, follow these steps:

1. From the SAS Model Manager window, select **Tools** ⇨ **Manage Templates**. The SAS Model Manager Template Editor appears.
2. Select **File** ⇨ **Open** and navigate to **`<drive>\Tutorial4\Samples`** and select **ScoreRangeTemplate.xml**. Click **OK**. The template opens in the Template Editor.

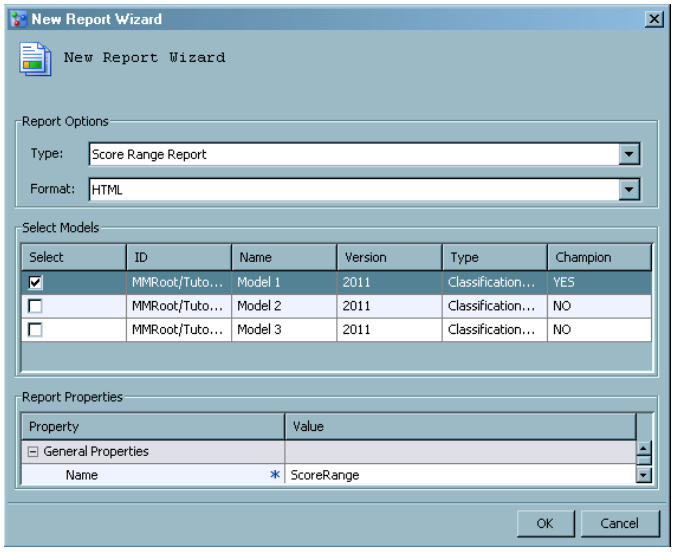



3. Select **File** ⇨ **Open** and navigate to **`<drive>\Tutorial4\Samples`**. In the **Files of type** box, select **SAS files (*.sas)**. Select **ScoreRange.sas** and click **OK**. The template opens in the Template Editor.
4. Select **File** ⇨ **Upload File**, verify the file information in the Upload File window, and click **OK**.
5. Select **File** ⇨ **Open** and navigate to **`<drive>\Tutorial4\Samples`**. In the **Files of type** box, select **SAS files (*.sas)**. Select **ScoreRangeMacro.sas** and click **OK**. The template opens in the Template Editor.
6. Select **File** ⇨ **Upload File**, verify the file information in the Upload File window, and click **OK**.

### *Run the New User-defined Report*

To execute the installed score range report, follow these steps:

1. Log on to SAS Model Manager.
2. Expand the **Tutorial2** folder, the **Delinquency** project, and the **2011** version.
3. Right-click the **Reports** folder and select **Reports** ⇨ **New Report Wizard**. The New Report Wizard opens.
4. In the **Type** box, select **Score Range Report**.
5. In the **Select Format** box, select **HTML**.
6. In the **Select Model** table, select **Model 1**.
7. If a ScoreRange report exists in the Reports folder from the previous exercise, you can name the report using the default filename. If **ScoreRange** does not exist in the **Reports** folder, type **`ScoreRange`** in the **Name** box of the **General Properties** table.




8. Click **OK**.
9. Click **Close** in the information message.
10. To view the new report, expand the new score range report, right-click **ScoreRange.html**, and select **Open**. To view the report output, see Display 5.1 on page 92 and Display 5.2 on page 93.

For more information about this task, see the *SAS Model Manager 3.1: User's Guide*.

*Chapter 6*

# Tutorial 5: Publishing Scoring Functions



## Overview of Publishing a Scoring Function

SAS Model Manager enables you to publish models that are associated with the **Data Step** score code type to a configured database. When you publish a scoring function for a project, SAS Model Manager exports the project's champion model to the SAS Metadata Repository. The SAS Scoring Accelerator then creates scoring functions in the default version that can be deployed inside the database based on the project's champion model score code. The scoring function can be validated automatically against a default train table to ensure that the scoring results are correct. A scoring application or SQL code can then execute the scoring functions in the database. The scoring functions extend the database's SQL language and can be used in SQL statements in the same way as other database functions. For more information, see “Publish Scoring Functions” in Chapter 13 of *SAS Model Manager: User's Guide*.

This tutorial shows you the tasks that are involved in publishing a project champion model to a database. It contains examples and step-by-step directions about preparing a database for use with SAS Model Manager and publishing a scoring function.

*Note:* The examples that are used in this tutorial are for publishing a scoring function to a Teradata database. You can also use this tutorial to publish a scoring function to a DB2, Greenplum, or Netezza database.

## Prerequisites

### *Models Used in Tutorial 3*

The exercises in this tutorial depend on some of the properties of the specific models that were added in Chapter 4, “Tutorial 3: Importing and Exporting Models,” on page 69.

Use the projects, versions, or models that are specified here. This tutorial is designed to follow Tutorial 3.

# Prepare a Database for Use with SAS Model Manager

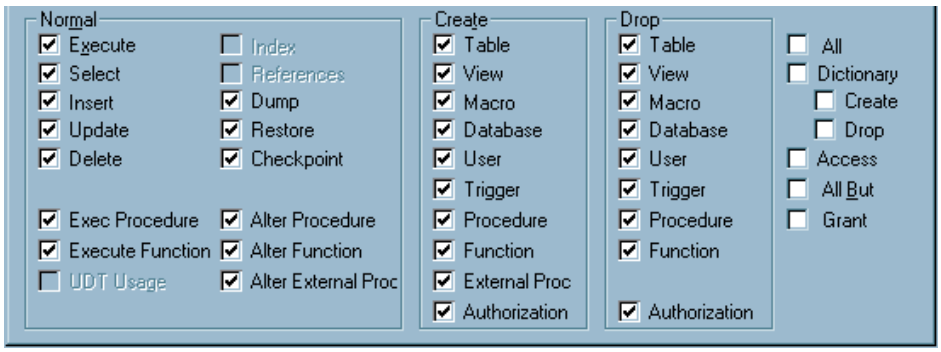
To use the SAS Model Manager Publish Scoring Function, the Database Administrator (DBA) needs to prepare the database. In this exercise, the DBA sets up the publishing and scoring aspects of SAS Model Manager.

The SQL scripts that are required for this tutorial are on your local computer after you extract them from the ZIP file SMM31Tutorial.zip. If you have not extracted the tutorial files, see "Install and Register the Tutorial Files" on page 3.

*Note:* Contact your system administrator if you do not have the appropriate permissions to the installation and configuration directories on the SAS Model Manager server.

To enable the publishing of scoring functions to a database from SAS Model Manager, follow these steps:

1. Create a database. You need to provide the server name, user ID, and password to users, so they can publish a scoring function from SAS Model Manager.
2. Set the user access permissions for the database that you created in the previous step. For example, grant all **Normal**, **Create**, and **Drop** permissions that are available in the Grant/Revoke Objects dialog box using the Teradata Administrator 12.0 or 13.0 tool. Here is an example of the permissions that are set for a Teradata database:




   *Note:* For more information about configuring a database, see "Preparing a Database for Use with SAS Model Manager" in Chapter 7 of *SAS In-Database Products: Administrator's Guide*.

3. Run the create tables script for the appropriate database to create the SAS Model Manager metadata tables in the database. For example, use the **createTablesTD.sql** script to create the metadata tables for a Teradata database. The SQL file is located in the **Utilities** subdirectory of the SAS Model Manager In-Database Scoring Scripts installation directory on the middle-tier server. Here is the default installation directory for a Microsoft Windows server:

   ```
   C:\Program Files\SASHome
   \SASModelManagerInDatabaseScoringScripts\3.1
   ```

4. (Optional) Run the **hmeq.sql** script to create the **hmeqid** table in the database that can be used with a scoring application. The SQL file is located in the`<drive>\Tutorial5` folder. To run the **hmeq.sql** file, follow these steps:

a. Start a BTEQ session.

b. Issue a login statement. For example:

```
.login myserver.com/mmtest,mmtest1
```

c. Set the scoring database as the active database in the BTEQ session. For example:

```
database mmtest;
```

d. Issue the command to run the SQL script. For example:

```
.run file="<drive>\Tutorial5\hmeq.sql"
```

5. Install the SAS 9.3 Formats Library for the database. This library contains many of the formats that are available in Base SAS.

   For information about how to install and configure the SAS 9.3 Formats Library, see the chapter on post-installation configuration for the SAS Accelerator Publishing Agent software in the *Configuration Guide for SAS 9.3 Foundation* for your operating environment.

6. Download the JDBC Driver. For example, download the jar files (**terajdbc4.jar** and **tdgssconfig.jar**) for Teradata 12.0 or Teradata 13.0 from the Teradata Web site (http://downloads.teradata.com/download) and place the jar files on the Web application server in the ***JBoss_Home*\server\SASServer1\lib** directory.

   *Note:* For information about where to place the jar files or other Web application servers, see "Preparing a Database for Use with SAS Model Manager" in Chapter 7 of *SAS In-Database Products: Administrator's Guide*.

7. Send a request to your system administrator to restart the Web application server to finish the installation of the JDBC jar files.

*Note:* For more information about preparing a database for use with SAS Model Manager, see the *SAS Model Manager: Administrator's Guide*.

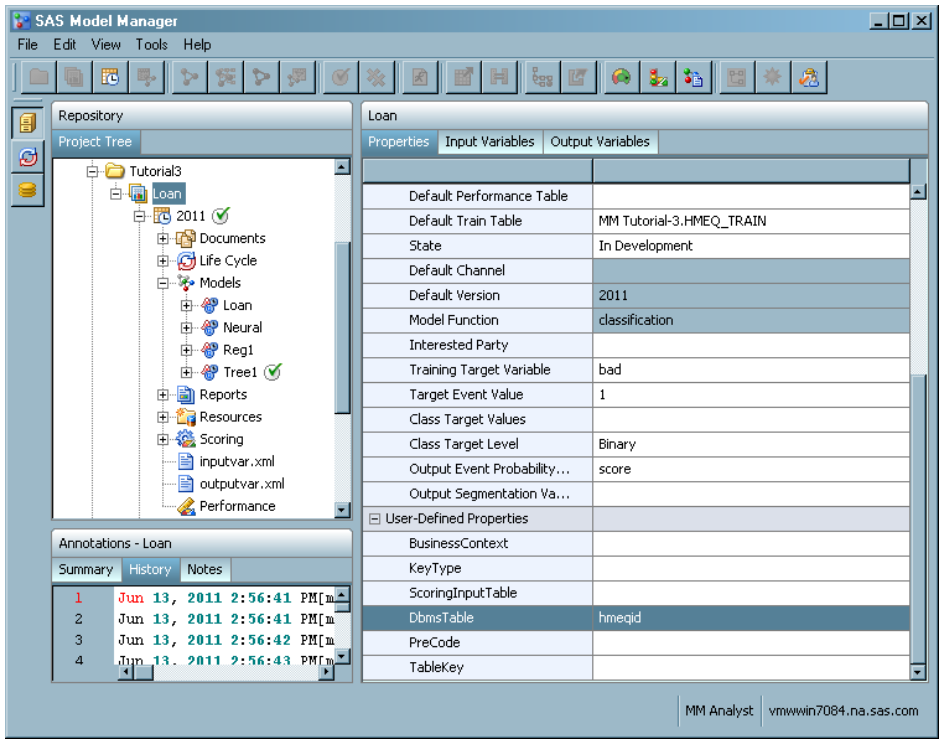
# Publish a Scoring Function

In this exercise, you publish a scoring function for a project's champion model to a database.

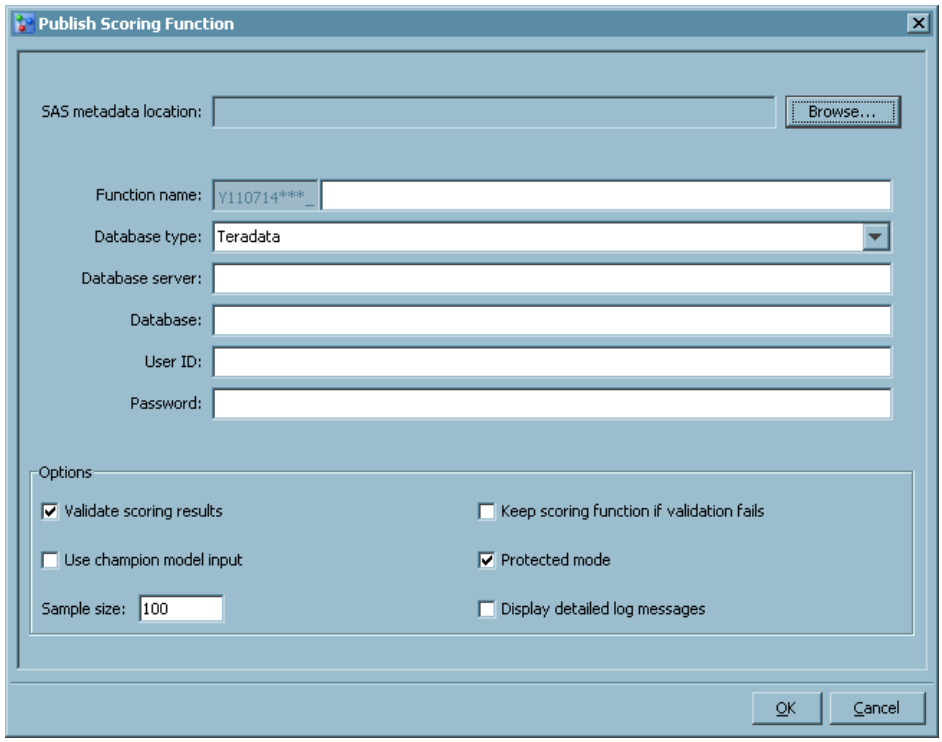
To publish a scoring function, follow these steps:

1. Verify that you have set the default version for the project and have set the champion model for the default version in Tutorial 3.

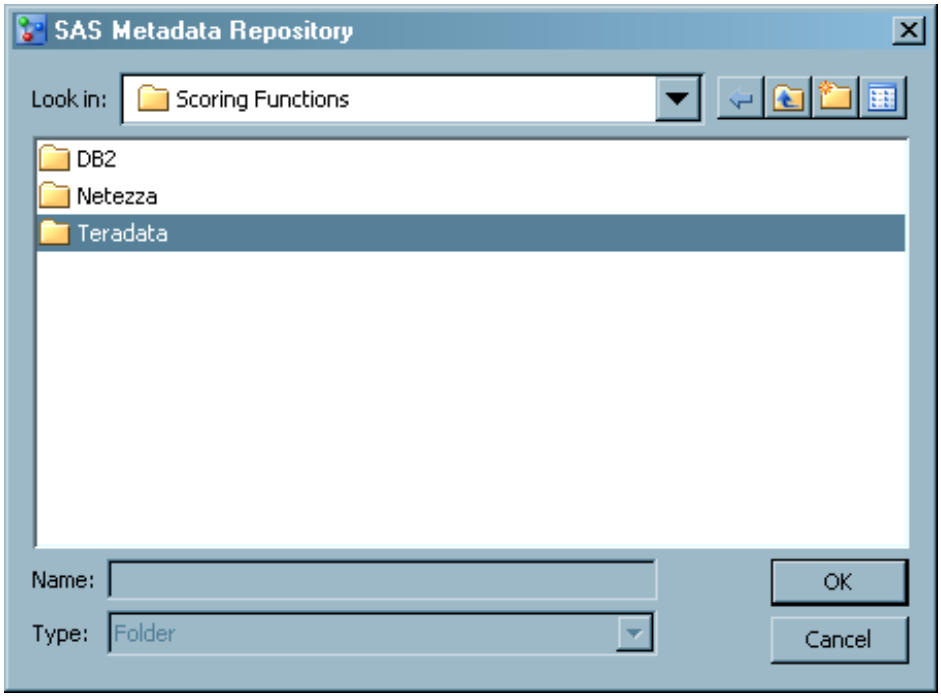
2. (Optional) Select the **Loan** project folder and enter a value for the **DbmsTable** user-defined property. This value is the scoring input table that the DBA might have created in the database to be used with a scoring application.




3. Right-click the **Loan** project in the Project Tree and select **Publish Scoring Function**. The Publish Scoring Function window appears.




4. Select a location in which to publish the model. Click **Browse**, select a folder name, and then click **OK**.

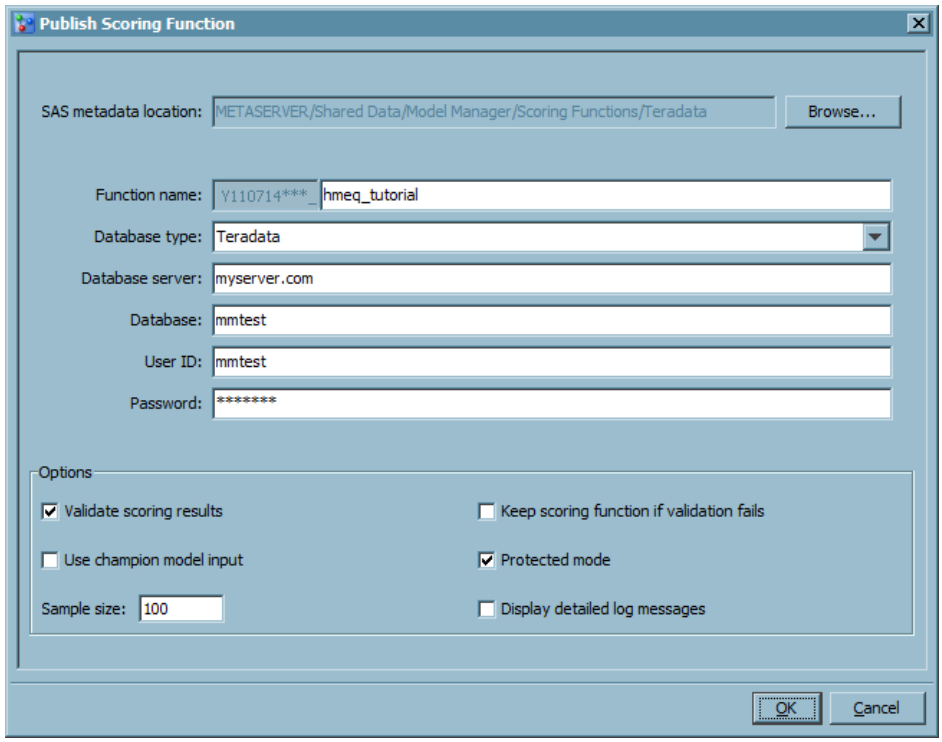



5. Enter a name for the scoring function, using the following naming conventions:
   - The user-defined value must be unique across all projects.
   - The user-defined value is case insensitive. The maximum length for the user-defined value is determined by which database type is selected, and no spaces are allowed. The maximum length of the user-defined value is 19 alphanumeric characters for Teradata, 117 alphanumeric characters for Netezza and DB2, and 52 alphanumeric characters for Greenplum.
   - The only special character that can be included in the function name is an underscore.
6. Select a database type from the drop-down list. The type of database that you choose determines which fields are displayed.
7. For the following fields, enter the values for the database that your DBA created for this tutorial:
   - **Database server**
   - **Database**
   - **User ID**
   - **Password**

   *Note:* If you select a database type other than Teradata, additional fields are required. For more information, see the *SAS Model Manager: User's Guide*.
8. In the **Options** section, select one of the following check boxes for the desired validation options.
   - **Validate scoring results**
   - **Keep scoring function if validation fails**
   - **Use champion model input**
   - **Protected mode**
   - **Display detailed log messages**

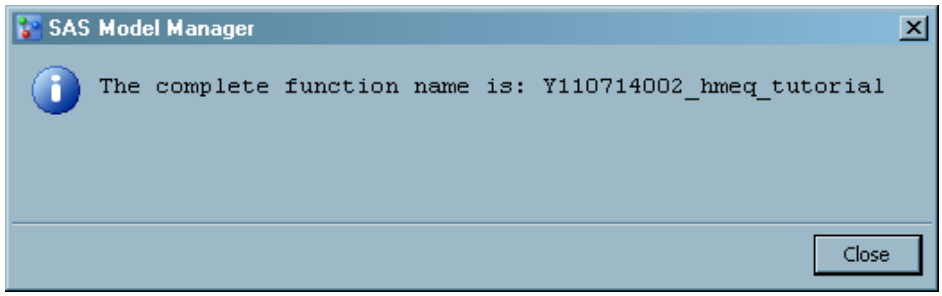
*Note:* By default, the **Validate scoring results** and **Protected mode** options are selected by default. The **Keep scoring function if validation fails** option is available for selection only when **Validate scoring results** is selected.

9. Enter a numeric value for **Sample Size**. The default sample size is 100 if the value is null or zero. The maximum number of digits that are allowed is 8.

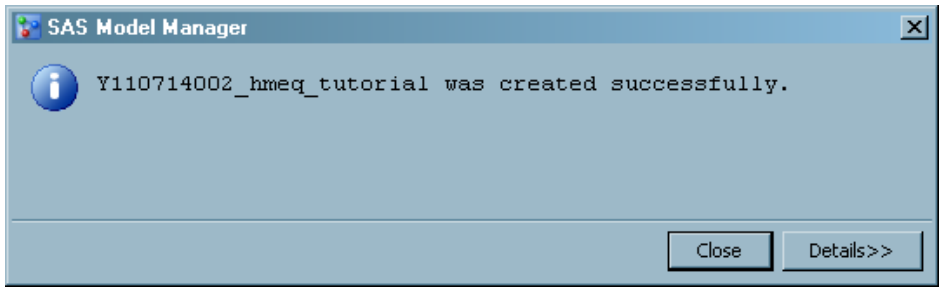



10. Click **OK**. A message is displayed that contains the scoring function name.

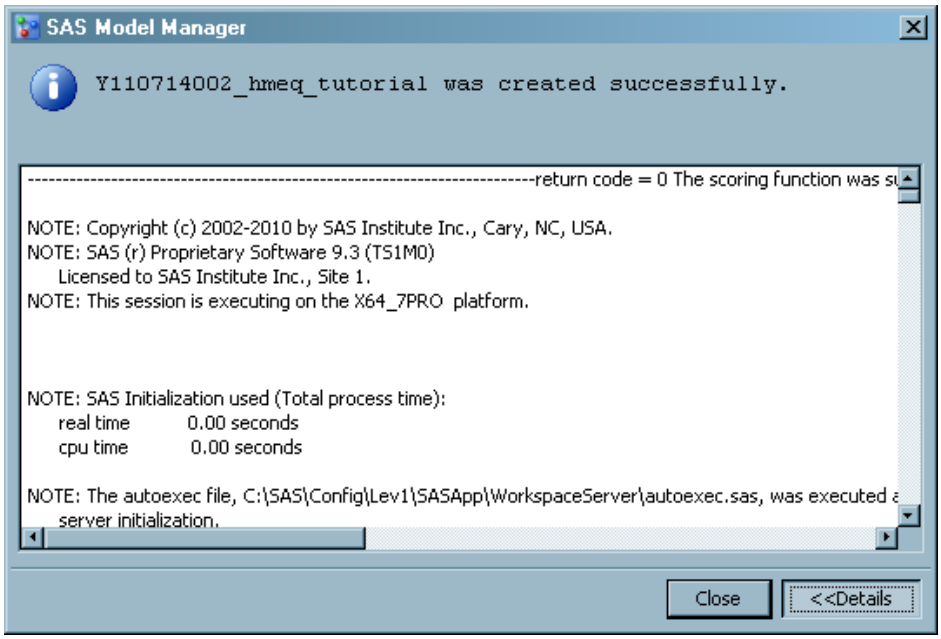
*Note:* The user-defined value of the **Function Name** is validated against the target database. If the user-defined value is not unique across projects, an error message is displayed.



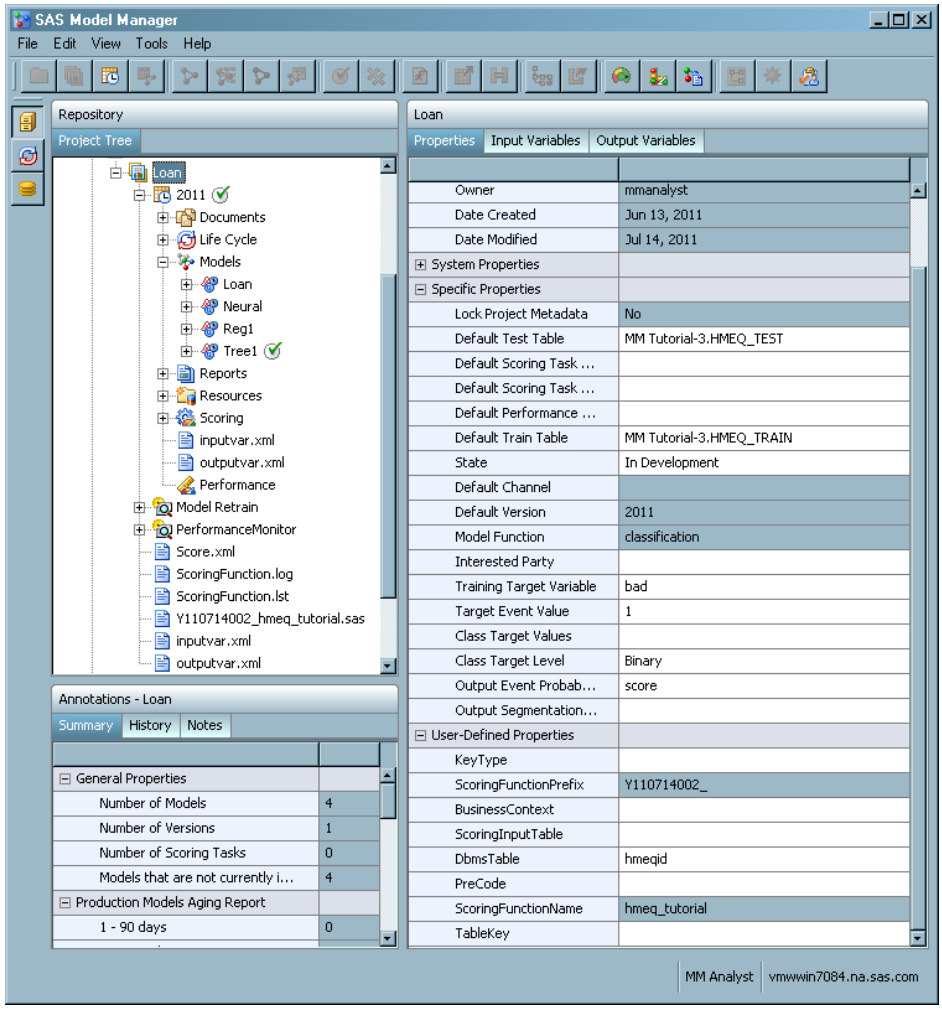

11. Click **OK**. A message indicating that the scoring function has been created successfully or unsuccessfully is displayed.



12. Click **Details** to display information about the publish scoring function actions or error messages.




13. Click **OK** to complete the publishing process. The SAS score code for the scoring function (for example, **`Y110714002_hmeq_tutorial.sas`**) is added to the project file list and the **ScoringFunctionName** and **ScoringFunctionPrefix** user-defined project properties are populated.




14. The actions that are performed during the publishing process are displayed in the history. To view the history of the project, select the project name and then click the **History** tab in the **Annotations - Loan** pane.

*Note:* After you complete the publishing process, you can view a log of the actions that were performed during the scoring function publishing process in the **`ScoringFunction.log`** file. The file is located in the project folder. For more information about the contents of the scoring function log, see the *SAS Model Manager: User's Guide*.

*Chapter 7*

# Tutorial 6: Using Advanced SAS Model Manager Features



## Overview of Using Advanced Features

This tutorial is designed to enable a user who is already familiar with the administrative and the basic functions of SAS Model Manager to perform the following tasks:

- create a new model template using the SAS Model Manager Template Editor
- upload the new model template to the SAS Content Server
- import a model that is described by the template
- generate a report on the model

The tutorial contains examples and step-by-step directions for performing these tasks.

## Example Scope

The model used in this tutorial is based on the ARBORETUM procedure, which is a SAS Enterprise Miner procedure.

# Prerequisites

The exercises in this tutorial require that the Tutorial 6 data sets and models from SMM31Tutorial.zip be extracted and registered in SAS Management Console. If they have not been extracted and registered, see "Prepare Tutorial 6 Data Sets and Models" on page 9 to extract and register the files.

The **<*drive*>\Tutorial6\Samples\Model6** folder contains these model template files that are used in this tutorial:

- importance6.sas7bdat
- modelinput6.sas7bdat
- modeloutput6.sas7bdat
- nodestat6.sas7bdat
- path6.sas7bdat
- rules6.sas7bdat
- score6.sas
- target6.sas7bdat

# Organize the Model Hierarchy

In this exercise, you use the Project Tree to create a modeling project.

## *Create a Folder*

To provide an organizational folder to manage your modeling projects, follow these steps:

1. Click the **Projects perspective** button  .
2. Right-click the **MMRoot** node in the Project Tree and select **New** ⇨ **New Folder**. The New Folder dialog box appears.
3. Specify the following folder properties and click **OK**.

   **Name**
   enter **Tutorial6**.

   **Description**
   enter an optional folder description.

## *Create a New Project*

To create a project and define the model function, follow these steps:

1. Right-click **Tutorial6** and select ⇨ **New** ⇨ **New Project**. The New Project dialog box appears.

2. Specify the following project properties and click **Next**:

   **Name**
   enter **`HmeqVars`**.

   **Description**
   enter an optional description.

   **Model Function**
   select **classification**.

3. Under the **Project Input Variables** table, click **Import Variables** and navigate to the **Tutorial6** folder in the SAS Metadata Repository. Select **HMEQ_PROJECT_INPUT** and click **OK**.

4. Under the **Project Output Variables** table, click **Import Variables** and navigate to the **Tutorial6** folder in the SAS Metadata Repository. Select **HMEQ_PROJECT_OUTPUT** and click **OK**.

5. Click **Finish**.

### *Define the Project Properties*

To define the properties that SAS Model Manager uses to create reports and model scores, follow these steps:

1. Select the **HmeqVars** project in the **Tutorial6** folder and expand **Specific Properties** in the right pane.

2. Specify the default data tables and model variables for the project:

   **Default Test Table**
   select **HMEQ_TEST**.

   **Default Scoring Task Input Table**
   select **HMEQ_SCORE_INPUT**.

   **Default Scoring Task Output Table**
   select **HMEQ_SCORE_OUTPUT**.

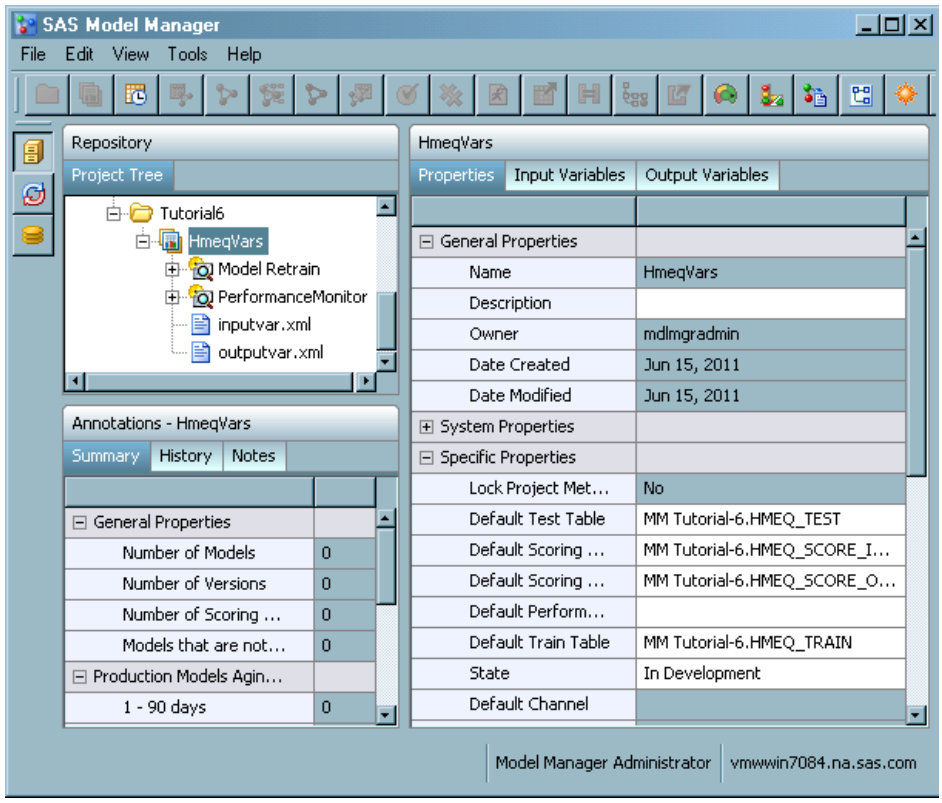
   **Default Train Table**
   select **HMEQ_TRAIN**.

   **Training Target Variable**
   enter **`bad`**.

   **Training Event Value**
   enter **`1`**.

   **Class Target Level**
   select **Binary**.

   **Output Event Probability Variable**
   select **score**.



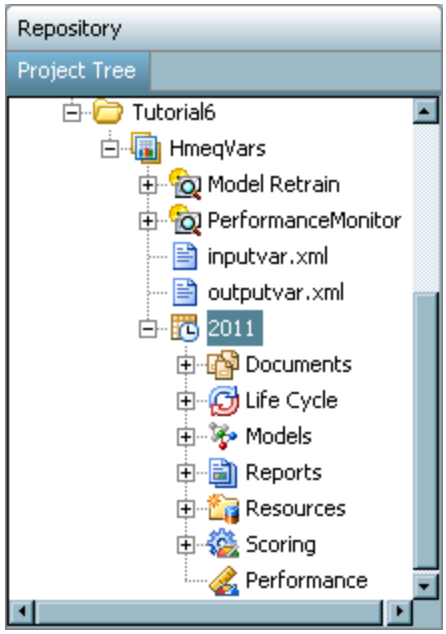

## Create a Version

Create the version for the project. The version folder contains life cycle information, auxiliary version documents, candidate model files, model comparison reports, resource files, scoring tasks, and model performance reports.

1. Right-click the **HmeqVars** project and select **New** ⇨ **New Version**. The New Version dialog box appears.
2. Specify the following version properties and click **OK**.

   **Name**
   enter `2011`.

   **Life Cycle Template**
   select the user-defined template **Tutorial Life Cycle** that you created in the first tutorial. For more information, see "Create a Life Cycle Template" on page 23. If **Tutorial Life Cycle** is not a selection in the list, select any life cycle template.
3. Examine the **HmeqVars** project to verify that it contains one version, **2011**.



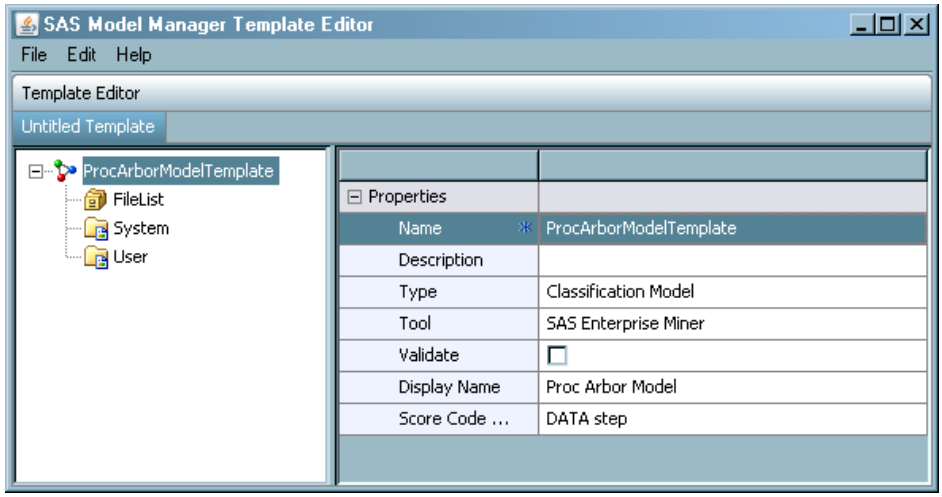

## Create and Upload a Model Template

In this exercise, you create a new model template using the SAS Model Manager Template Editor. For information about creating a model template, see the *SAS Model Manager: User's Guide*.

To create a model template, follow these steps:

1. Open the SAS Model Manager Template Editor. Select **Tools** ⇨ **Manage Templates**.
2. Open an empty model template in the template editor. Select **File** ⇨ **New Model Template**. An empty, untitled model template opens.
3. Specify the model template properties.
   a. In the **Name** field, replace **Untitled Template** with **`ProcArborModelTemplate`**.
   b. Click the **Type** box and select **Classification Model**.
   c. In the **Tool** field, enter **`SAS Enterprise Miner`**.
   d. In the **Display Name** field, enter **`Proc Arbor Model`**.
   e. Click the **Score Code Type** box and select **Data step**.

   Here is the template editor after this step has been completed:




4. Add the model component files and the model component file properties.

   The following table lists the model component files that comprise the model, and the properties for each file. For each model component file, add an entry under **FileList**. Then, select the file under **FileList** and enter the properties for that file. A value of **`none`** indicates that you do not need to set a value for that property.

   To add the model component files, right-click **FileList** and select **New File Item**. In the **Name** field, enter the name from the table and click **OK**.

   Here is the template after adding the file score.code as a model component file:

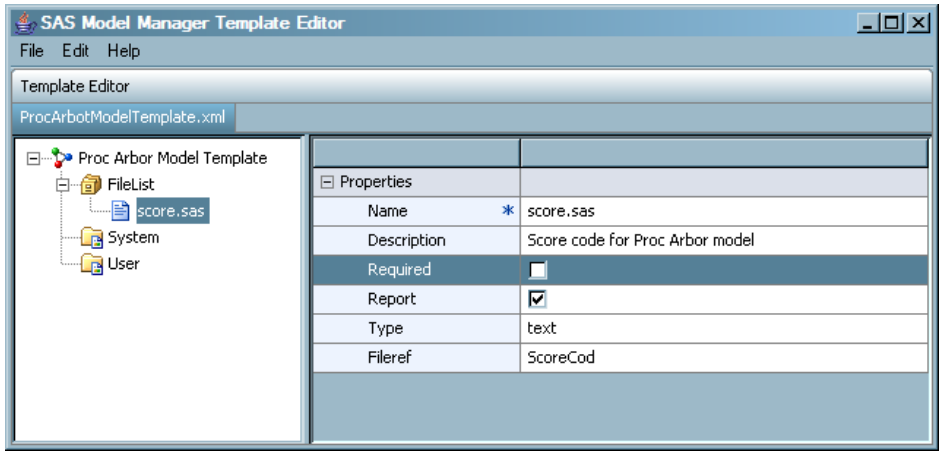



***Table 7.1*** *ProcArborModelTemplate Component Files and Component File Properties*

| Filename | Description | Required | Report | Type | Fileref |
|---|---|---|---|---|---|
| score.sas | Score code for Proc Arbor model | none | select the box | text | ScoreCod |
| modelinput.sas7bdat | Model input table | none | select the box | binary | none |
| modeloutput.sas7bdat | Model output table | none | select the box | binary | none |
| inputvar.xml | Input variable XML file | none | select the box | text | Input |

| Filename | Description | Required | Report | Type | Fileref |
|---|---|---|---|---|---|
| outputvar.xml | Output variable XML file | none | select the box | text | Output |
| target.sas7bdat | Target variable table | select the box | select the box | binary | Target |
| importance.sas7bdat | Variable relative importance | none | select the box | binary | none |
| path.sas7bdat | Path information | none | select the box | binary | none |
| rules.sas7bdat | Node rules | none | select the box | binary | none |
| nodestat.sas7bdat | Node statistics | none | select the box | binary | none |

5. Add a system property.

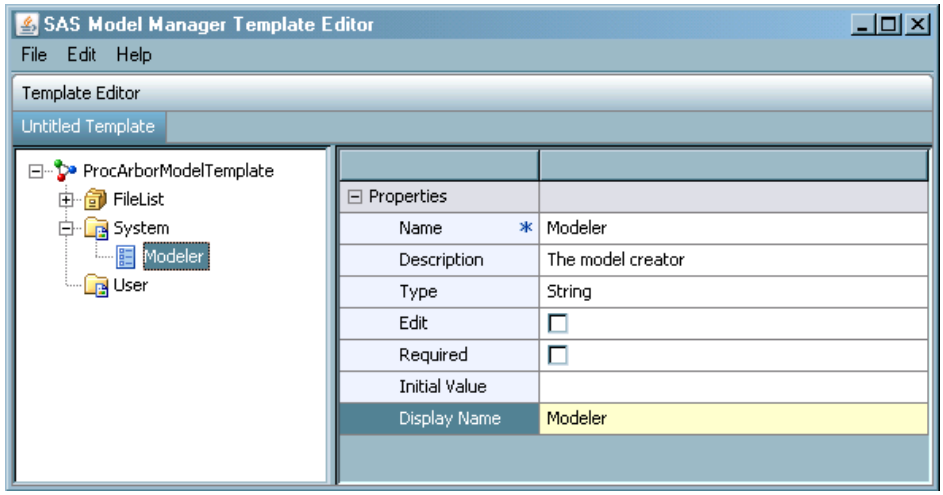
   a. In the left pane, right-click **System** and select **New Property**. In the **Name** field, enter `Modeler` and click **OK**.

   b. Select **Modeler** and enter the following property values:

   **Description**
   `The model creator.`

   **Display Name**
   `Modeler`

   Default values are used for all other property values. Here is the template after the system property has been added:




6. Add user properties.

   The following table lists user properties for the model template and the properties of each user property. For each user property, add an entry under **User**. Then, select the property under **User** and enter the properties for the user property. A value of none indicates that you do not need to set a value for that property.

   To add a user property, right-click **User** and select **New Property**. In the **Name** field, enter the name from the table and click **OK**.

Here is the template after the user property **Citi1** has been added:

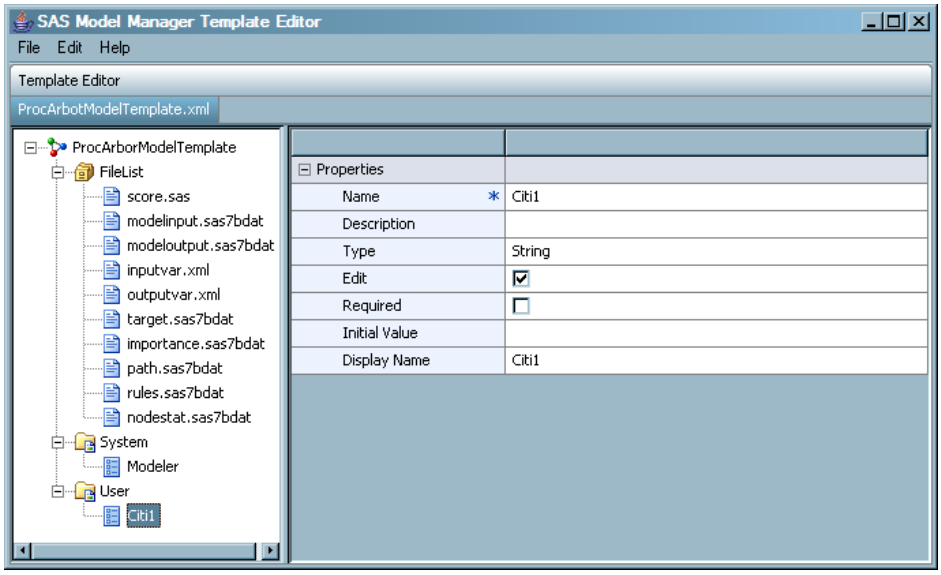
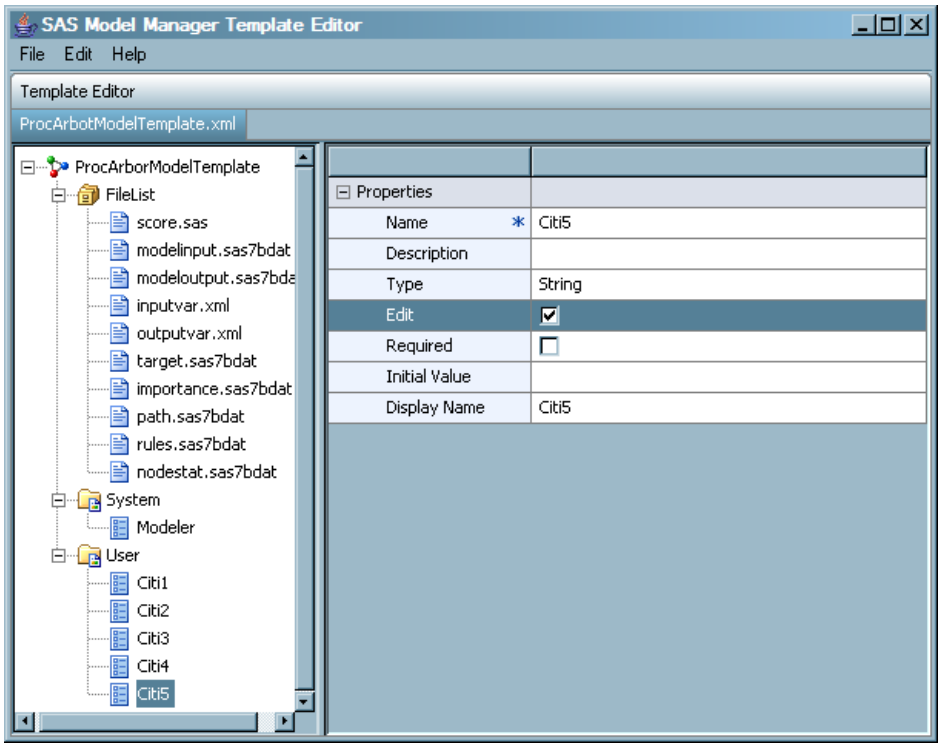



***Table 7.2*** *ProcArborModelTemplate User Properties and User Property Properties*

| Filename | Description | Type | Edit | Required | Initial Value | Display Name |
|---|---|---|---|---|---|---|
| Citi1 | none | String | select the box | none | none | Citi1 |
| Citi2 | none | String | select the box | none | none | Citi2 |
| Citi3 | none | String | select the box | none | none | Citi3 |
| Citi4 | none | String | select the box | none | none | Citi4 |
| Citi5 | none | String | select the box | none | none | Citi5 |

7. Save the template. Saving the template creates a backup of the template. Select **File** ⇨ **Save As** and enter **`ProcArborModelTemplate.xml`** in the **File name** field. Click **OK**.

   Here is the model template after all files and properties have been added to the template:



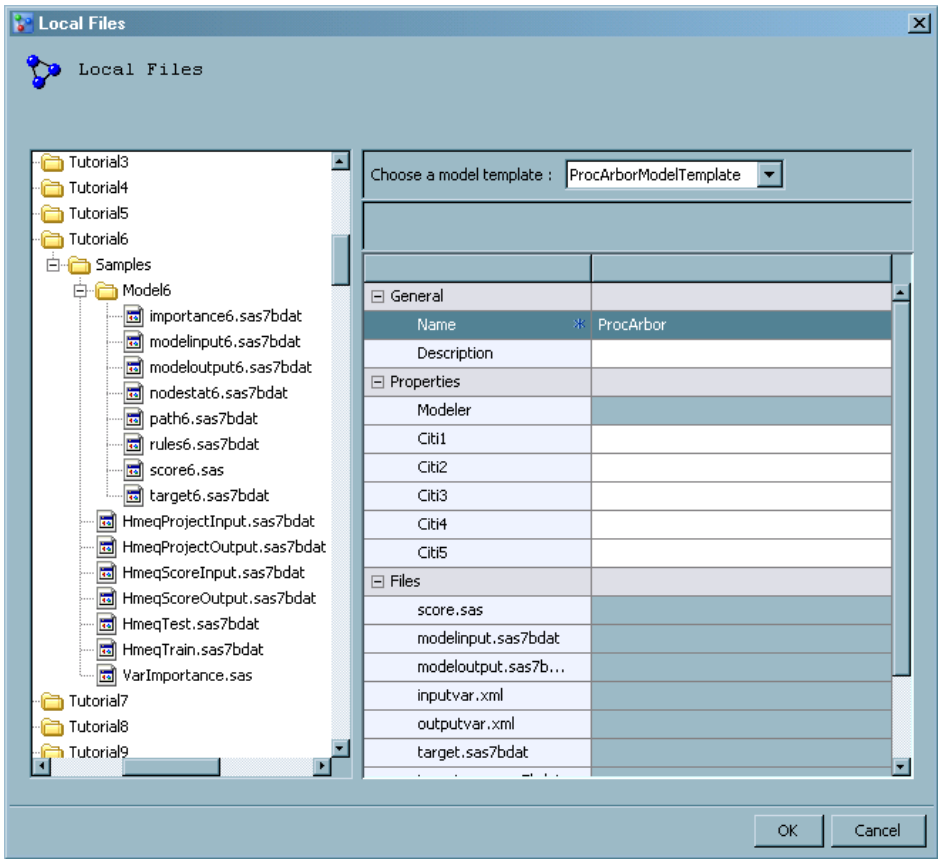

8. Select **File** ⇨ **Upload File** to upload the template to the SAS Content Server. In the Upload File window, verify the information and click **OK**.
9. Select **File** ⇨ **Exit** to close the SAS Model Manager Template Editor.

# Import a Model

## *Import a Model*

In this exercise, you import a model using the user model template:

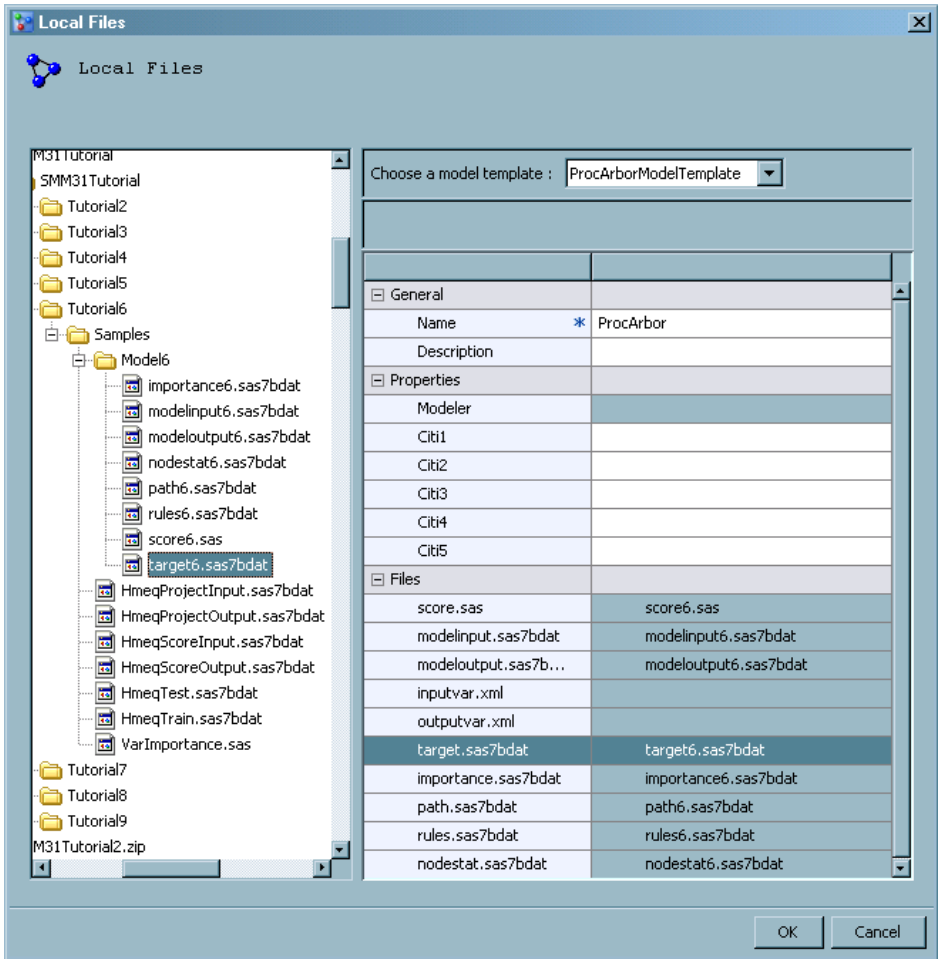
1. Log on to SAS Model Manager.
2. In the **2011** version of the **HmeqVars** project, right-click **Models** and select **Import from** ⇨ **Local Files**.
3. From the **Choose a model template** box, select **ProcArborModelTemplate**. The custom properties and files appear in the right pane.
4. In the Name field, enter **ProcArbor**.
5. In the left pane, expand **Desktop** to **`<drive>:\Tutorial6\Samples\Model6`**. The Local Files dialog box displays the template in the right pane and the files to import in the left pane:



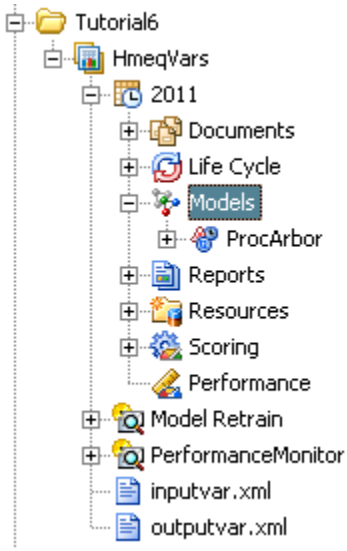

6. From the left pane, drag the following files to the corresponding fields in the model template:

| Filename | Template Field Name |
|---|---|
| **importance6.sas7bdat** | **importance.sas7bdat** |
| **modelinput6.sas7bdat** | **modelinput.sas7bdat** |
| **modeloutput6.sas7bdat** | **modeloutput.sas7bdat** |
| **nodestat6.sas7bdat** | **nodestat.sas7bdat** |
| **path6.sas7bdat** | **path.sas7bdat** |
| **rules6.sas7bdat** | **rules.sas7bdat** |
| **score6.sas** | **score.sas** |
| **target6.sas7bdat** | **target.sas7bdat** |

Here is the Local Files dialog box after the files have been assigned:




7. Click **OK**. The **ProcArbor** model appears under the **Models** folder.



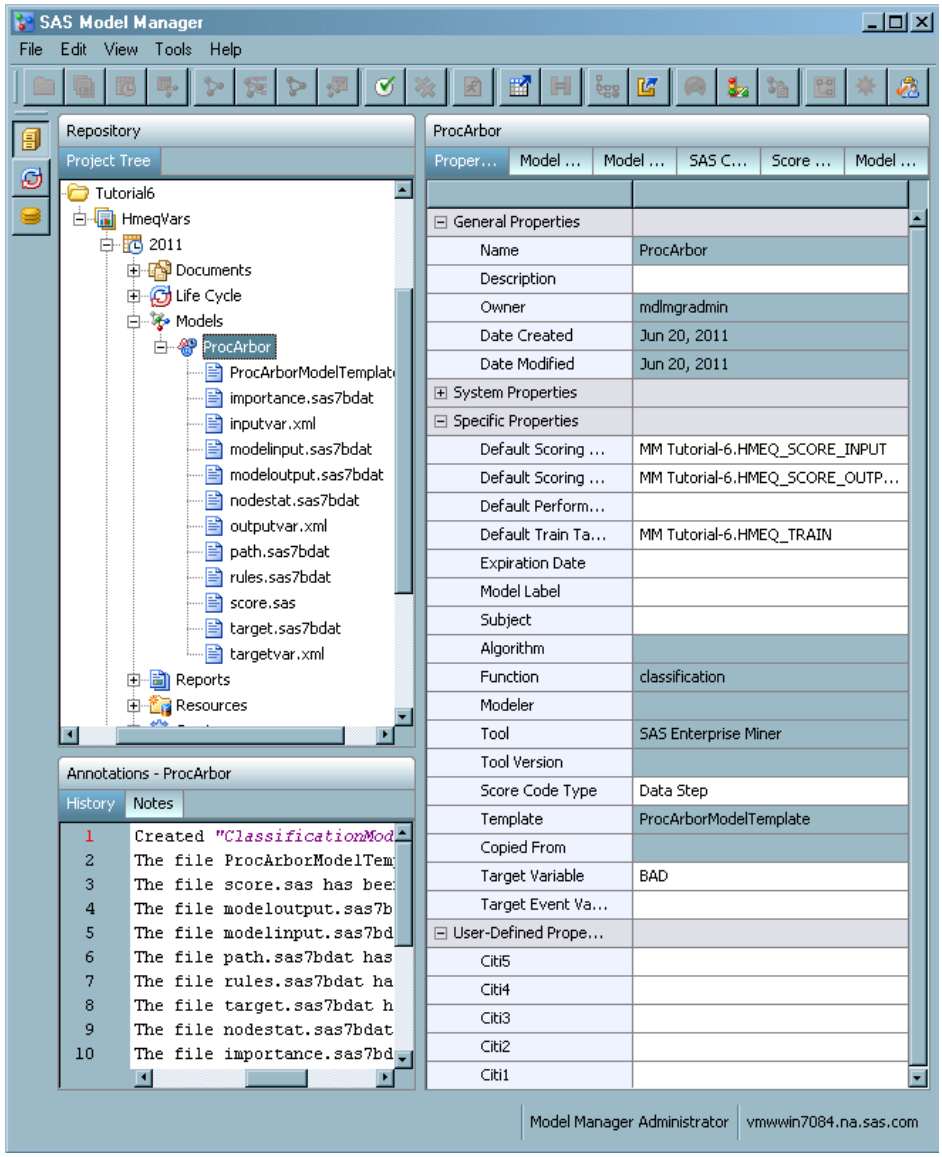

### *Verify Model Properties*

Verify some model properties:

1. In the Project Tree, expand the **Models** folder and select **ProcArbor**.

2. In the Properties View, verify **BAD** as the value of the **Target Variable** property.
3. Verify that the **Score Code Type** property is **Data Step**.

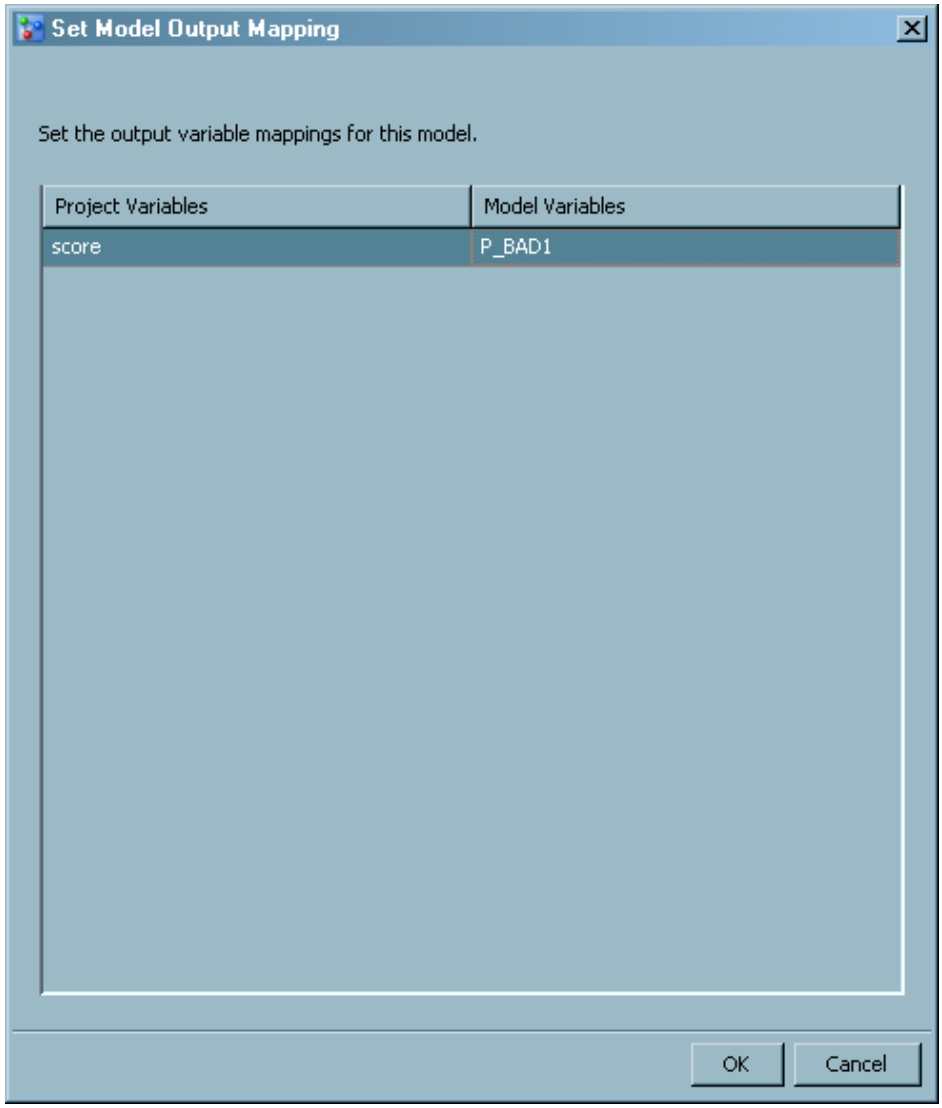
Here is the ProcArbor model in the SAS Model Manager window:




## *Map Model Output Variables to Project Output Variables*

Because the project output variable name is not the same as the model output variable name, the output variables must be mapped. To map the variables, follow these steps:

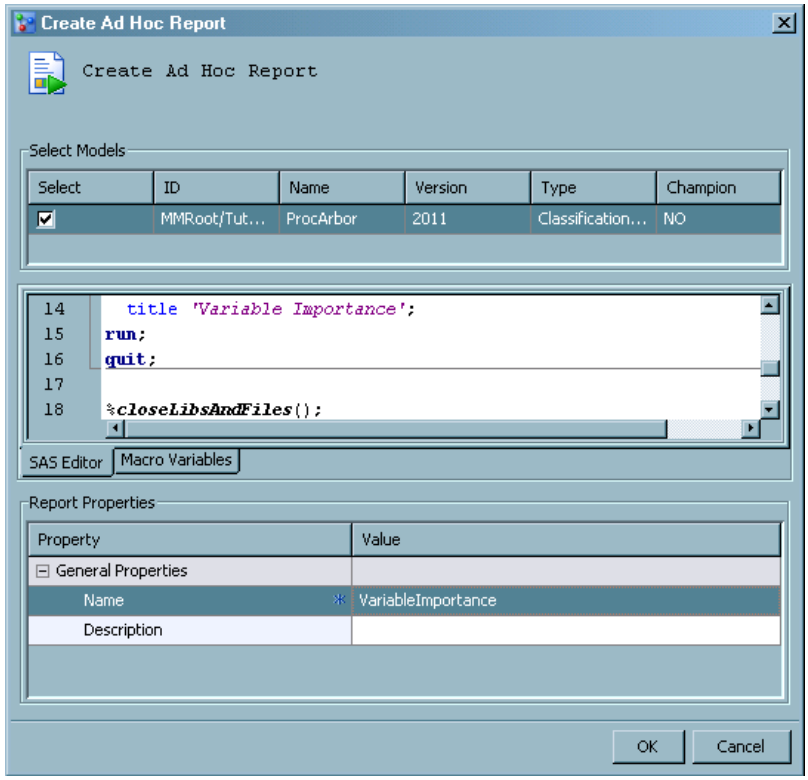
1. Right-click the model name, **ProcArbor**.
2. Select the **Set Model Output Mapping** option to open the Set Model Output Mapping window.
3. Click the **Model Variables** column for score, select **P_BAD1**, and click **OK**.




## Create an Ad Hoc Variable Importance Report

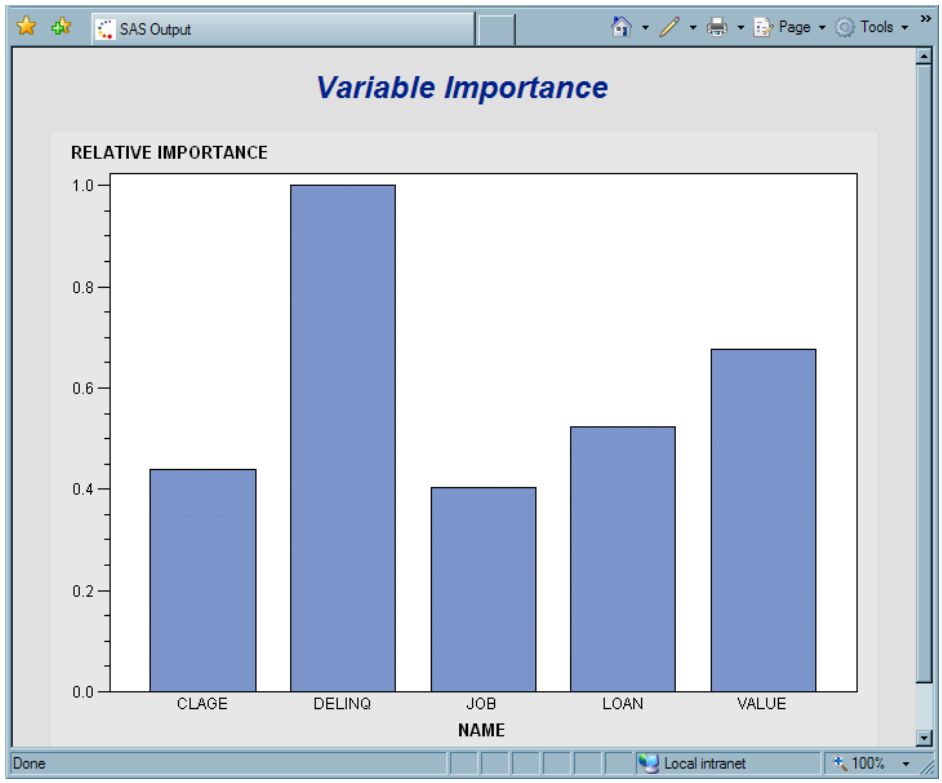
In this exercise, you create a report that is based on the model's PROC ARBORETUM importance data.

1. In the **<*drive*>:Tutorial6/Samples** folder, open the example report VarImportance.sas in a text editor and copy the code.
2. In the Project Tree, expand the **Tutorial6** folder and the project **HmeqVars**.
3. Under version **2011**, right-click the **Reports** folder and select **Reports** ⇨ **Create Ad Hoc Report**.
4. In the Create Ad Hoc Report dialog box, check the box for **ProcArbor** in the **Select Models** table.
5. In the SAS Editor, paste the code that you copied in Step 1.
6. In the **Name** field, enter **VariableImportance** and click **OK**. After the report is created successfully, click **Close** to close the information message.




7. View the report output.
   a. Expand the new report, **VariableImportance**.
   b. Right-click **ProcArbor.html** and select **Open**.
   c. If prompted, enter your user ID and password.

Here is the VariableImportance report:


SAS Output
Page
Tools
Variable Importance
RELATIVE IMPORTANCE
1.0
0.8
0.6
0.4
0.2
0.0
CLAGE
DELINQ
JOB
LOAN
VALUE
NAME
Done
Local intranet
100%

*Chapter 8*

# Tutorial 7: Creating Performance Monitoring Reports and Using Dashboard Reports



## Overview of Monitoring Reports

SAS Model Manager monitoring reports enable you to monitor and evaluate model performance. Model performance can sometimes be improved by tuning or refitting the model, or by using a new champion model.

To create monitoring reports, you create a performance task by using the Define Performance Task Wizard. Then, you execute the performance task. The output from executing a performance task includes several charts, including Characteristic, Stability, Lift, ROC and Gini, and Kolmogorov-Smirnov charts. You can view these charts in SAS Model Manager or you can create monitoring reports in PDF, HTML, RTF, or Excel output formats.

# Prerequisites

The exercises in this tutorial depend on some of the properties of the specific models that were added in Chapter 4, "Tutorial 3: Importing and Exporting Models," on page 69.

The performance data sets from SMM31Tutorial.zip must be extracted and registered in SAS Management Console. If the data sets have not been extracted and registered, see "Prepare Tutorial 7 Data Sets and Models" on page 11 to extract and register the files.

The **<*drive*>\Tutorial7\Samples** folder contains these performance data sets that are used in this tutorial:

- hmeq_2010Q2.sas7bdat
- hmeq_2010Q3.sas7bdat
- hmeq_2010Q4.sas7bdat
- hmeq_2011Q1.sas7bdat

# Create the Champion Model Performance Data Sets

In this exercise, you run the Define Performance Task wizard to create a performance monitoring task for the champion model, Reg1. The performance monitoring task uses the information that you supply in the Define Performance Task wizard to create SAS programs. You then execute the SAS programs to create the performance monitoring data sets.

## *Ensure the Project Properties Are Set*

The Define Performance Task wizard requires that three project properties be set before you can run the wizard. Expand the **Tutorial3** folder and click the **Loan** project. Ensure that the following project properties are set:

| Project Property | Value |
|---|---|
| **Training Target Variable** | **bad** |
| **Target Event Value** | **1** |
| **Output Event Probability Variable** | **score** |

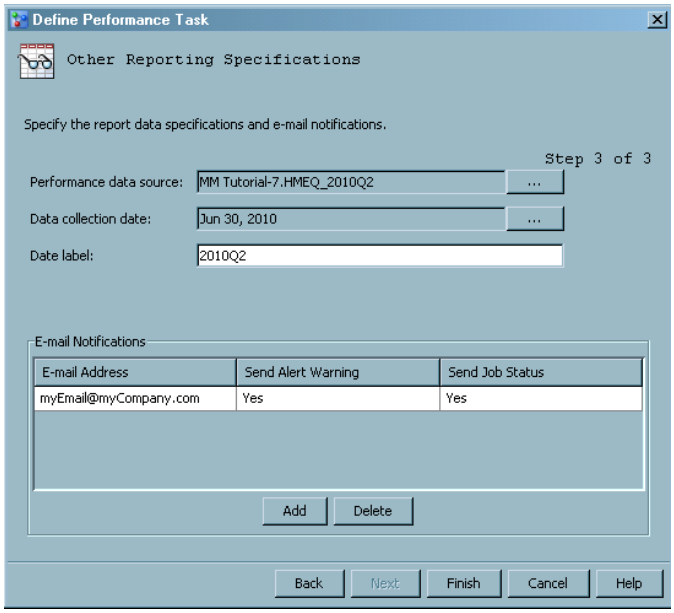
## *Run the Define Performance Task Wizard*

To run the Define Performance Task wizard, follow these steps:

1. Expand the **Tutorial3** organizational folder, right-click **Loan**, and select **Define Performance Task** from the pop-up menu. The Define Performance Task wizard appears.
2. In the **Output Variables for Stability Analysis** table, select the box for the **score** variable.
3. In the **Input Variables for Characteristic Analysis** table, click **Select All**. Click **Next**.
4. On the **Warning and Alert Conditions** page, accept the default alert and warning conditions by clicking **Next**. The **Other Reporting Specifications** page appears.
5. Click ... for the **Performance data source** box and select **HMEQ_2010Q2**. Click **OK**.
6. Click the ellipsis button for **Data collection date** and select **June 30, 2010** using the calendar. Click **OK**.

   *Note:* The date can be any date within the second quarter of 2010.
7. In the **Date label** box, enter `2010Q2`.

   *Note:* The label is used to identify the performance data in the performance monitoring charts. When you view the charts by using the Performance node in the Project Tree, SAS Model Manager uses the label **baseline** for the first set of performance data that is created for a champion model, and not the text that you entered in the **Date label** box.
8. Click **Add**. The Add Contact dialog box appears. Enter your e-mail address, and click **OK**. Here is the Define Performance Task wizard.

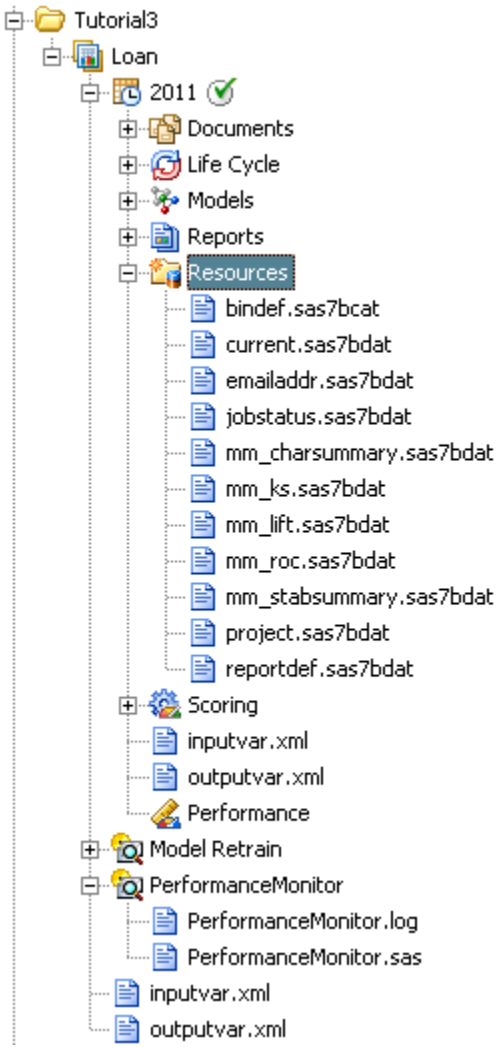



9. Click **Finish**. The wizard creates the SAS code that can be run to create the performance monitoring data sets.
10. Execute the SAS program. Under the **Loan** project, right-click **PerformanceMonitor** and select **Execute**. SAS Model Manager executes the performance monitoring program. When the program execution is complete, an information message indicates whether the program ran successfully. Click **Close**.

11. Expand **PerformanceMonitor**. Here you can see the SAS program that created the performance monitoring data sets and the resulting SAS log. Click both files to see the file contents in the **Content** pane.

12. Expand **Resources** under the default version **2011**. The **Resources** node contains the data sets that are created by the Define Performance Task wizard as well as the data sets that are created by executing the performance task. When a performance task is executed the first time for a given champion model, the performance task creates the initial data sets that are used for plotting the model performance charts. In executing subsequent performance tasks that use new performance data for the given champion model, SAS Model Manager appends the resulting data sets to the existing data sets. All of the data in the model performance data sets for a given champion model is used to plot the model performance charts.

    Click on any file to see the contents of that file in the **Content** pane.




13. Select **Performance**. The Performance node displays the champion model performance data as a graph and as a data set.

    *Note:* To view at least one line segment in Characteristic and Stability graphs, SAS Model Manager requires performance data sets from three performance task executions, at a minimum.

14. Define a performance task and execute the SAS program for the remaining three Tutorial 7 performance data sources. Complete steps 1 through 10 for each performance data source.

    On the Define Performance Task wizard, page 1, select all input and output variables if they are not already selected.

    On page 2, use the default alert and warning conditions. No changes are necessary.

    On page 3, use these values for the **Performance data source**, **Data collection date**, and **Date label** boxes:

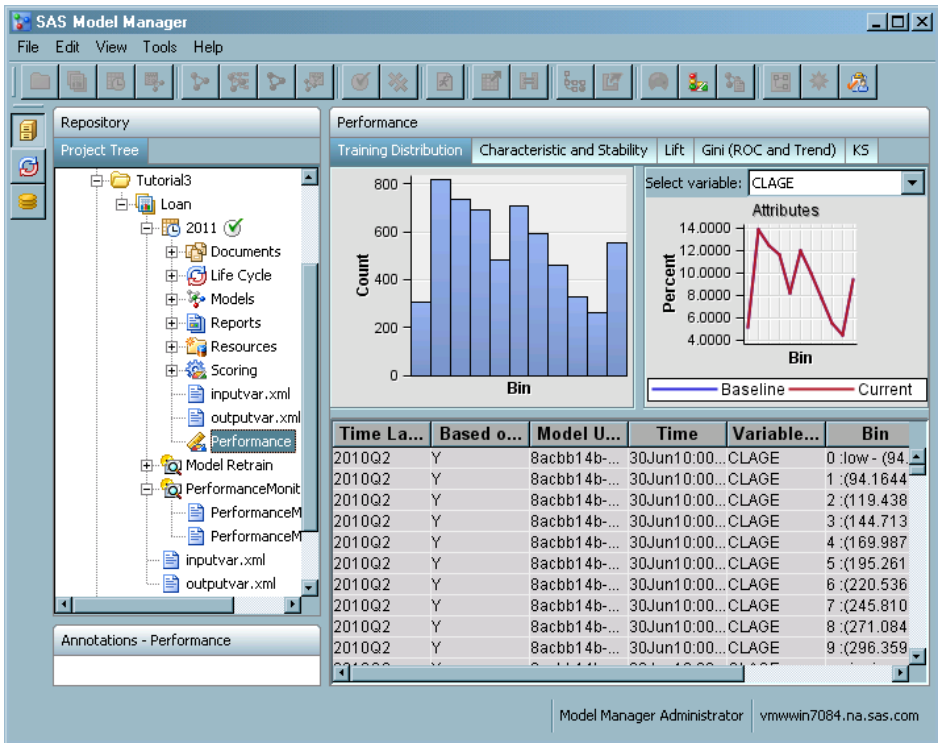
| Performance data source | Data collection date | Date label |
|---|---|---|
| HMEQ_2010Q3 | September 30, 2010 | 2010Q3 |
| HMEQ_2010Q4 | December 31, 2010 | 2010Q4 |
| HMEQ_2010Q1 | March 31, 2011 | 2011Q1 |

# View Performance Charts

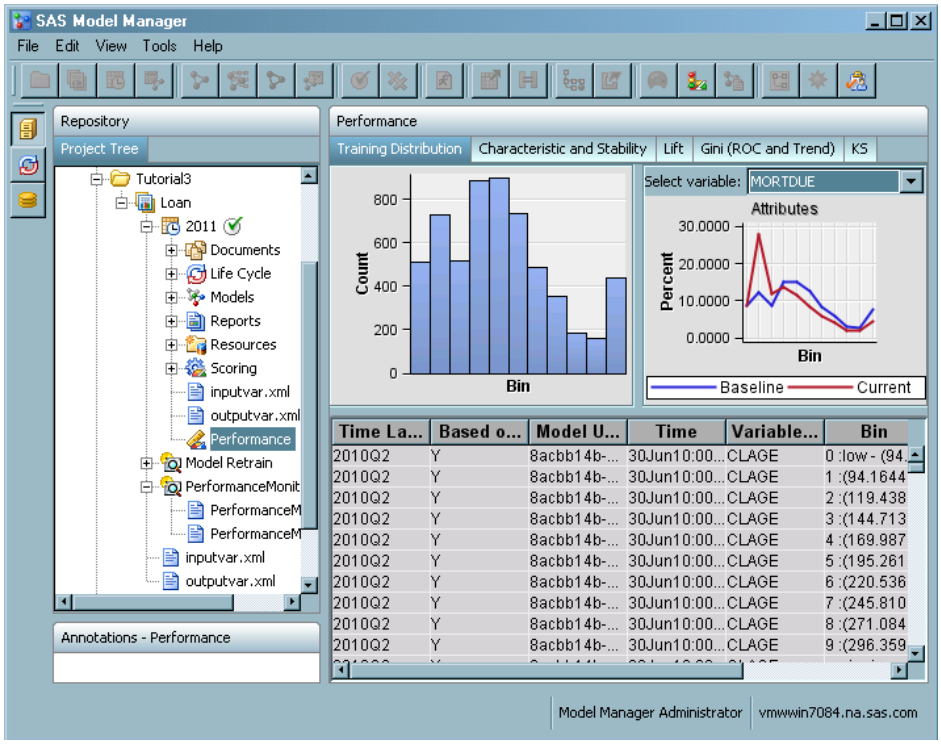
## *View the Training Distribution Chart*

To demonstrate the Training Distribution chart features, follow these steps:

1. In the SAS Model Manager Project Tree, expand **Tutorial3**, expand **Loan**, and expand **2011**.
2. Select the **Performance** node to display the Performance charts.

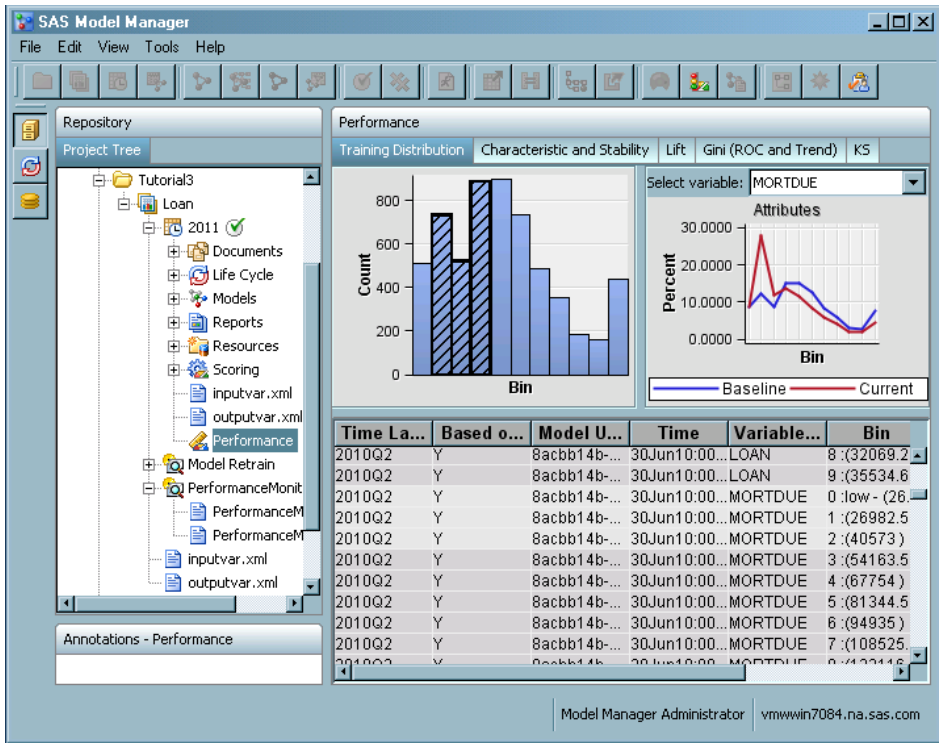



3. On the **Training Distribution** tab, click the **Select variable** box and select **MORTDUE**. The training distribution data and charts display the data for the MORTDUE variable.




4. Click on a bar to highlight the corresponding entry in the table. If necessary, move the scroll bar to locate the highlighted entry in the table.

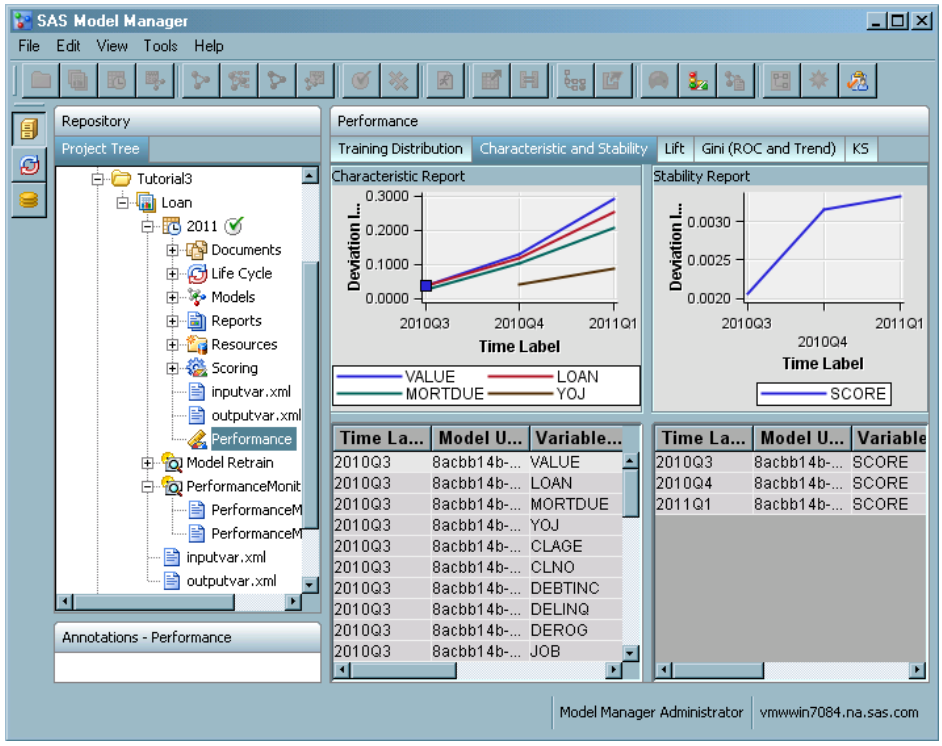
5. In the table, click the highlighted row and drag the mouse to select multiple rows. The associated bars in the chart are highlighted as you select the associated rows.




## *View the Characteristic and Stability Charts*

To demonstrate the Characteristic and Stability chart features, follow these steps:

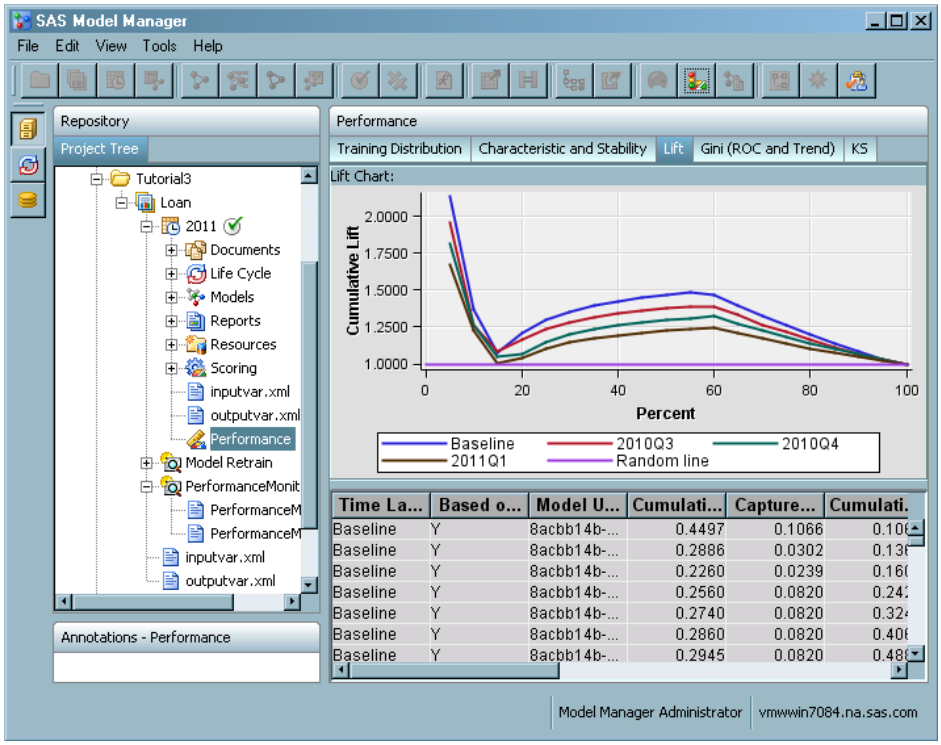
1. Select the **Characteristic and Stability** tab.
2. Select table entries to highlight the corresponding chart points.



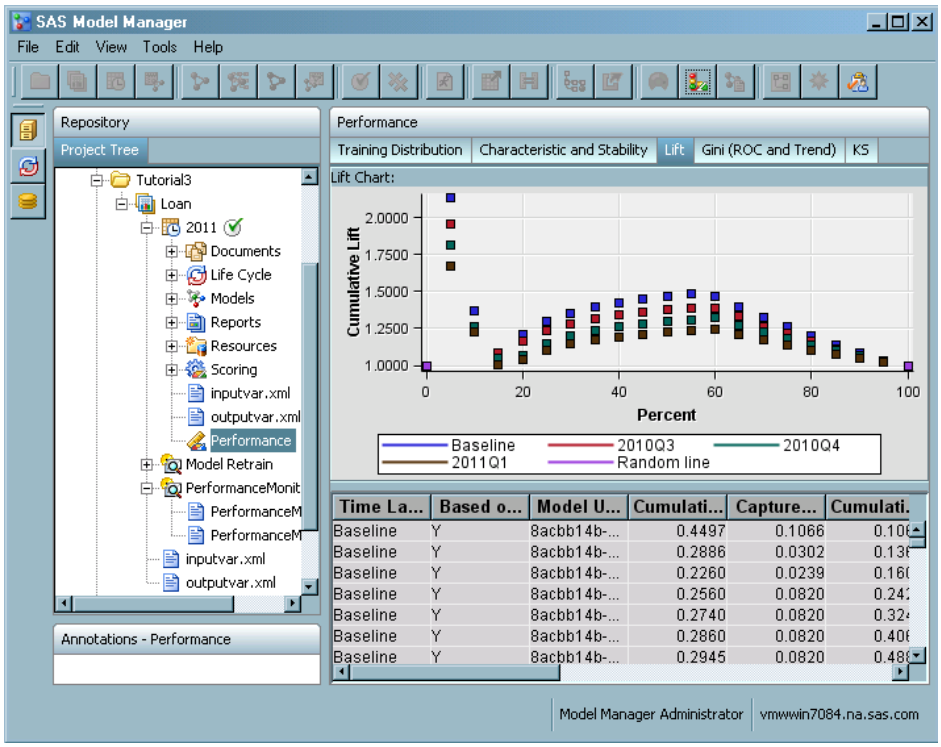
### View the Lift Chart

To demonstrate the Lift chart features, follow these steps:

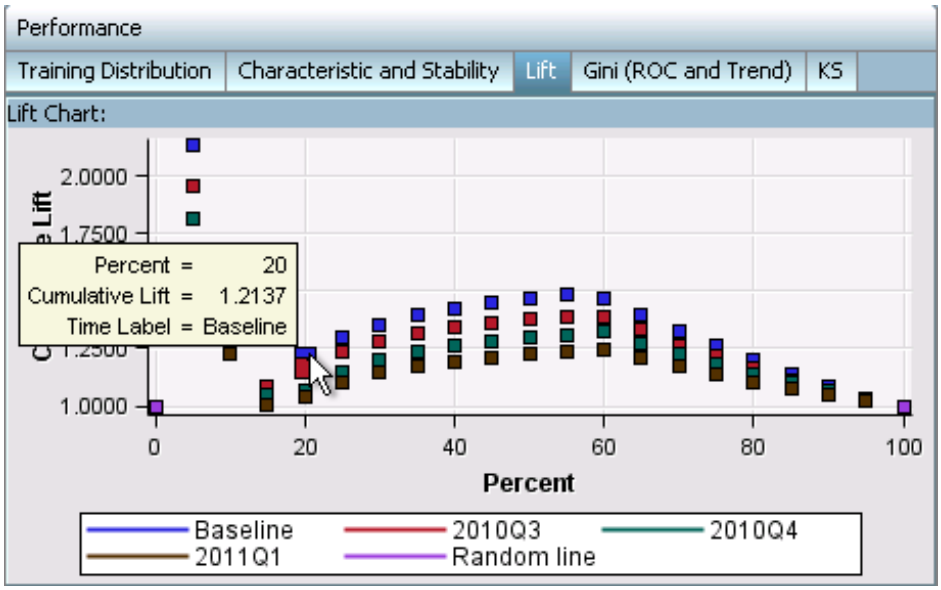
1. Select the **Lift** tab.




2. To change the chart to a plot chart, right-click the chart and select **Chart Type** ⇨ **Plot**



3. Move the pointer along one lift plot. You should see a pop-up box when the pointer is resting on a data point or is close to a data point.

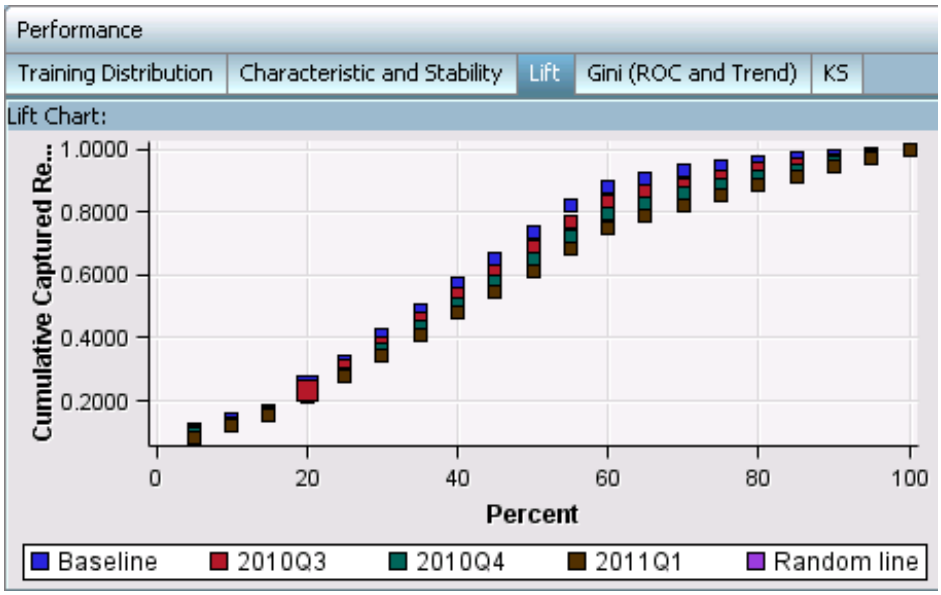



4. To chart the cumulative captured response:

   a. Right-click the chart area.

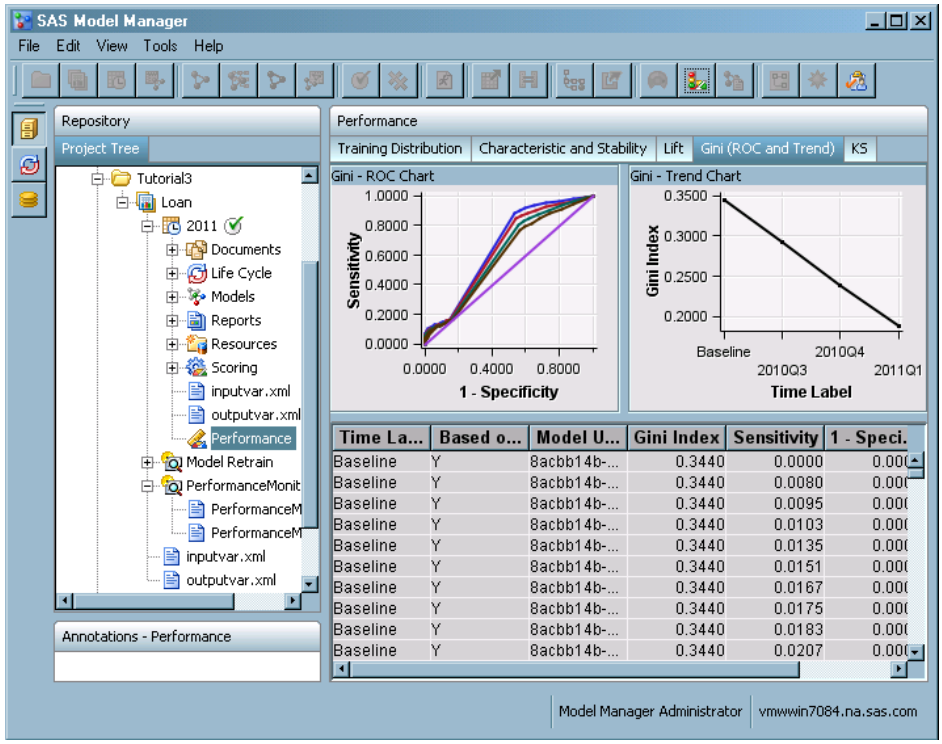
   b. Select **Data Options**. The Data Options Dialog window appears.

5. Click the **Roles** column for the variable **cuCapturedResp** and select **Y**. Click **OK**.




### *View the Gini (ROC and Trend) Charts*

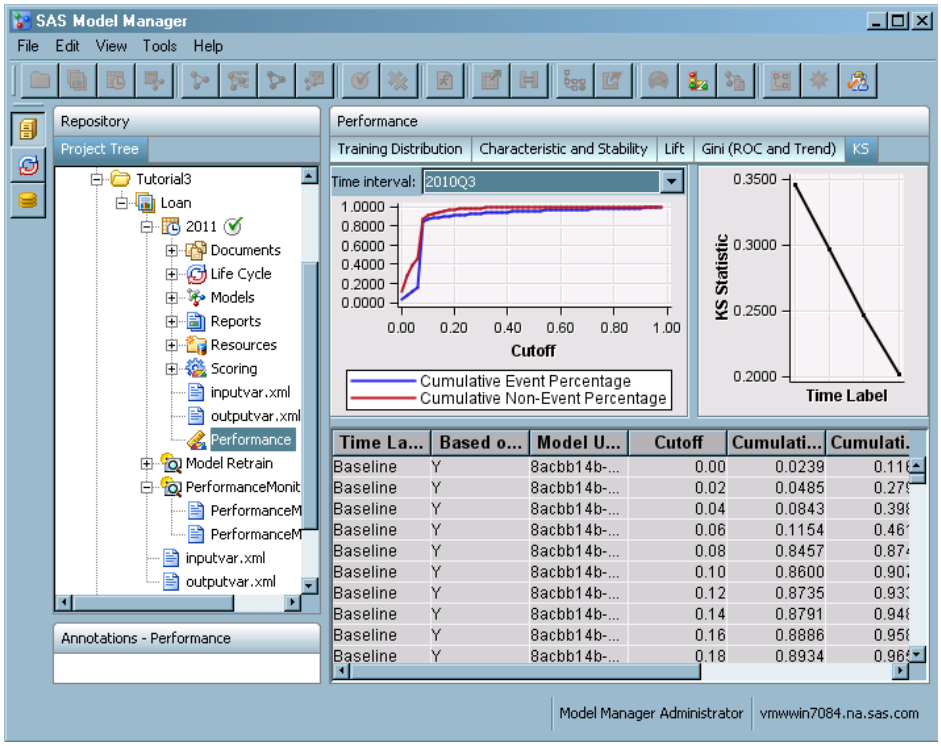
To view the ROC and Gini charts, select the **GINI (ROC and Trend)** tab.




## *View the Kolmogorov-Smirnov (KS) Chart*

To view the KS chart from the SAS Model Manager user interface, follow these steps:

1. Select the **KS** tab.

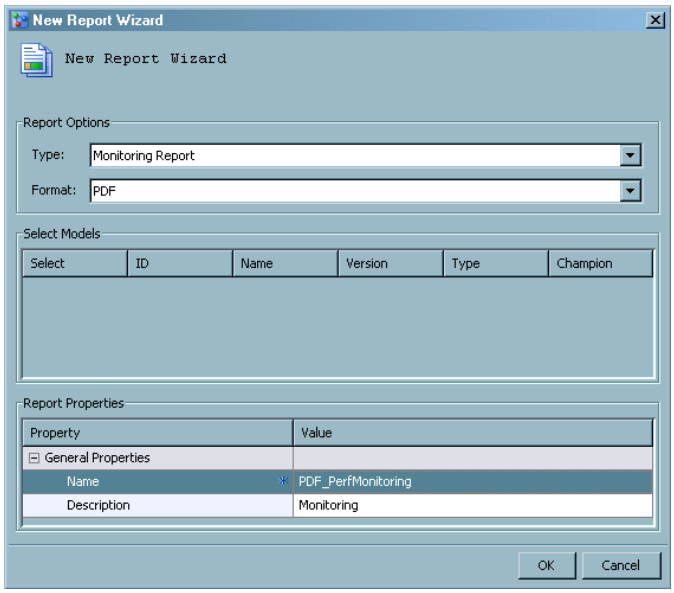



2. Select a different time point from the drop-down list of the **Time Interval** field.

# Creating Output Formats for Performance Monitoring Reports

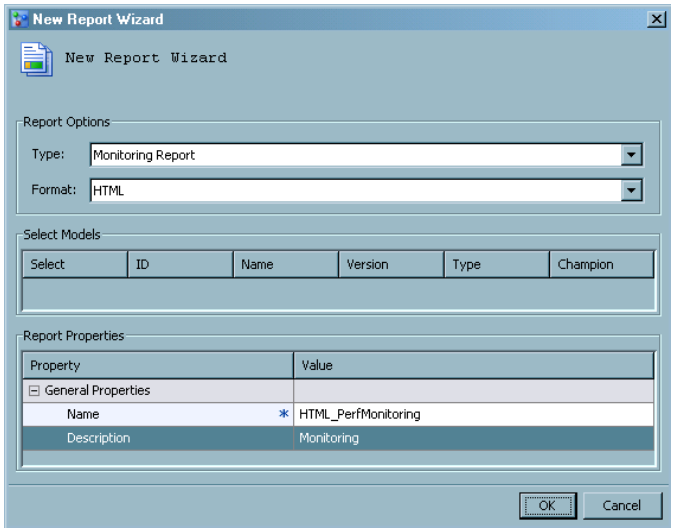
In this exercise, you use the New Report Wizard to create the monitoring reports in PDF and HTML output formats.

1. In the Project Tree, expand **Tutorial3**, the **Loan** project, and the version **2011**.
2. Right-click **Reports** and select **Reports** ⇨ **New Report Wizard**.
   a. Click the **Type** box and select **Monitoring Report**.
   b. In the **Name** box of **General Properties**, enter **`PDF_PerfMonitoring`**.




   c. Click **OK**. An information message indicates whether the report creation was successful. Click **Close** to close the message box.
   d. View the PDF performance monitoring report. Expand the **Reports** folder. Right-click **PDF_PerfMonitoring** and select **Reports** ⇨ **View Report**. Scroll through the report or click a link in the table of contents to view various parts of the report.
3. Create the same report in HTML.

   Right-click **Reports** and select **Reports** ⇨ **New Report Wizard**.
   a. Click the **Type** box and select **Monitoring Report**.
   b. Click the **Format** box and select **HTML**.
   c. In the **Name** box of the **General Properties**, enter **`HTML_PerfMonitoring`**.




d. Click **OK**. An information message indicates whether the report creation was successful. Click **Close** to close the message box.

e. View the HTML performance monitoring report. Expand the **Reports** folder. Right-click **HTML_PerfMonitoring** and select **Reports** ⇨ **View Report**. All charts and data appear on a single HTML page. Scroll through the report to view various parts of the report.

## Customize Model Monitoring Reports

In this exercise, you add a customized title to the performance monitoring report by modifying the SAS code that SAS Model Manager created in order to format the performance monitoring data.

1. In the **Reports** folder, expand **HTML_PerfMonitoring** and click **taskCode.sas**. The report program appears in the Content pane.
2. Scroll to the bottom of the report code. Select and copy the following code:

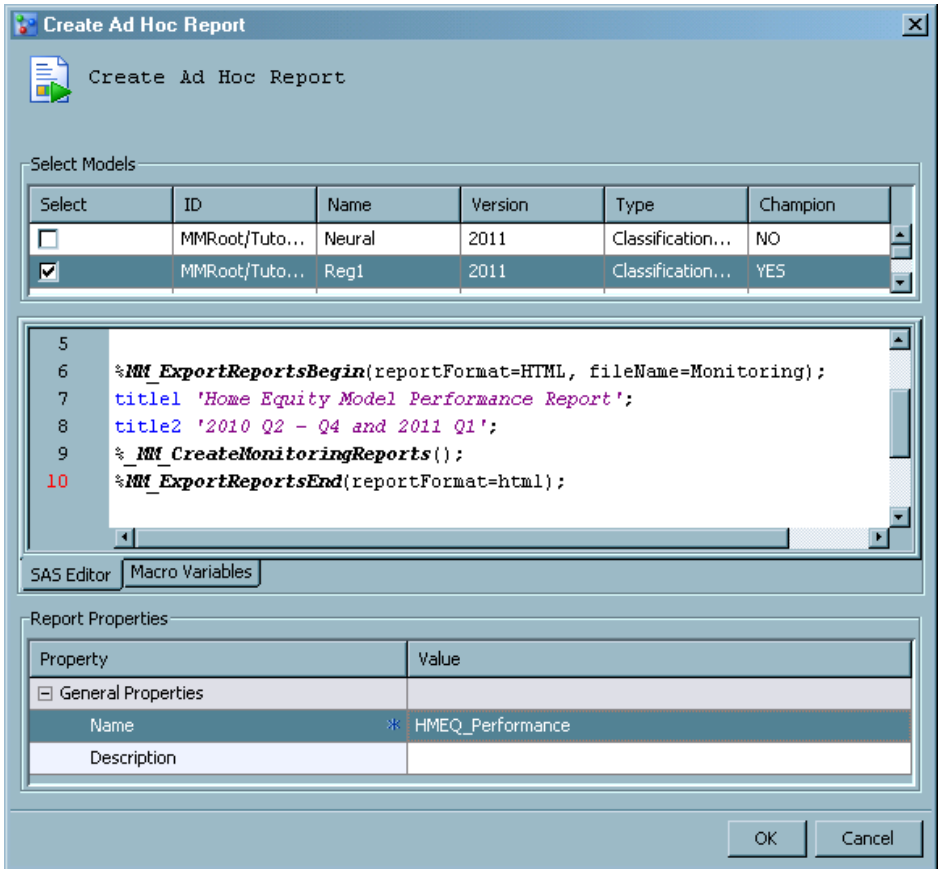
```
Filename mmreport catalog "sashelp.modelmgr.createmonitoringreports.source";
%include mmreport;
Filename mmreport catalog "sashelp.modelmgr.reportexportmacros.source";
%include mmreport;

%MM_ExportReportsBegin(fileName=Monitoring);
%_MM_CreateMonitoringReports();
%MM_ExportReportsEnd;
```

3. Right-click **Reports** and select **Reports** ⇨ **Create Ad Hoc Report**.
4. Select the box for the **Reg1** model.
5. In the SAS Editor, paste the code that you copied.
6. Modify these lines of code so that they match the following code:

```
%MM_ExportReportsBegin(reportFormat=html, fileName=Monitoring);
title1 'Home Equity Model Performance Report';
title2 '2010 Q2 - Q4 and 2011 Q1';
```

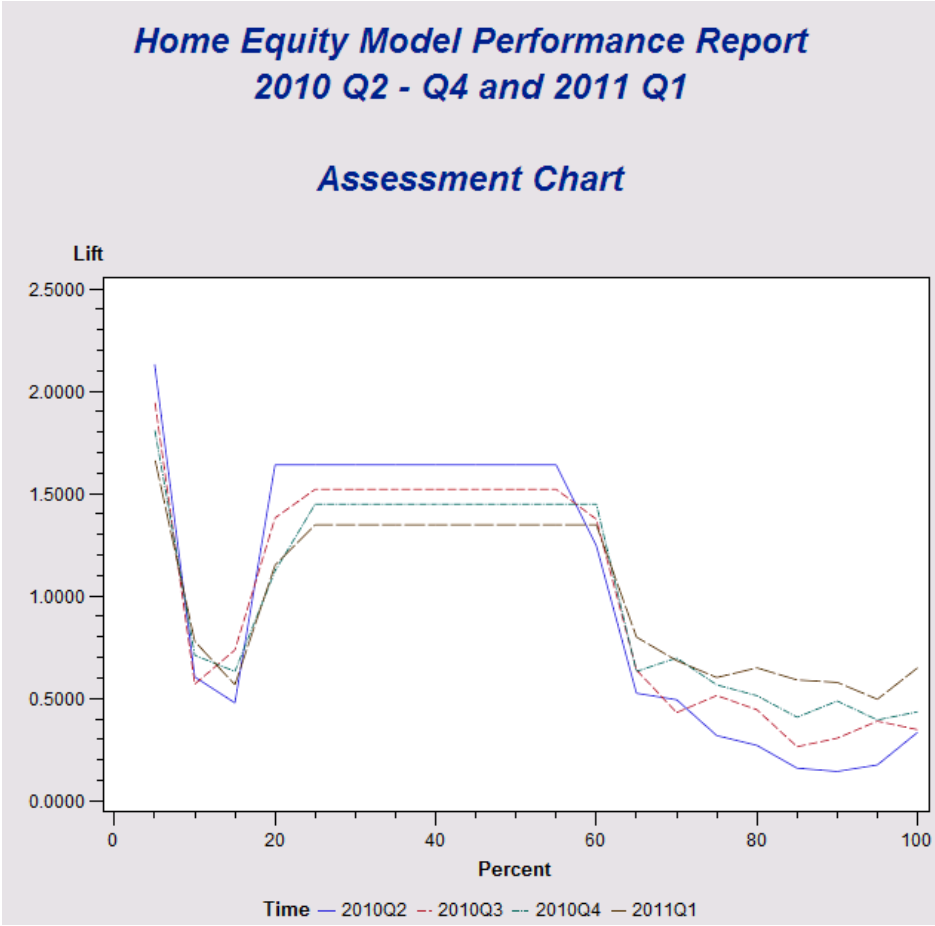
```
%_MM_CreateMonitoringReports();
%MM_ExportReportsEnd(reportFormat=html);
```

7. In the **Name** box, enter **`HMEQ_Performance`**.




8. Click **OK**. An information message indicates whether the report was successfully created. Click **Close** to close the message box.

9. In the **Reports** folder, expand **HMEQ_Performance**, right-click **Monitoring.html**, and select **Open**. Each chart has the new title. Here is the Lift chart:




## Using Dashboard Reports

### *Overview*

The SAS Model Manager Dashboard can provide reports that show the overall state of all projects that are being monitored. The dashboard reports are produced from existing performance monitoring reports. For each project, a user can define dashboard report indicators that are then used to create the dashboard reports. The dashboard reports are not displayed through the SAS Model Manager user interface. Instead, use a browser window to view the dashboard reports that are located on the SAS Workspace Server. These reports are generated in HTML by SAS Model Manager.

*Note:* The dashboard reports can be defined and generated only by SAS Model Manager administrators and advanced users.

In this exercise, you will define dashboard report indicators, generate the dashboard report, and view the dashboard reports.

## Prerequisites

### Models Used in Tutorial 3

The exercises in this tutorial depend on some of the properties of the specific models that were added in Tutorial 3. Use the projects, versions, or models that are specified here. This tutorial is designed to follow Chapter 4, "Tutorial 3: Importing and Exporting Models," on page 69.

### The Required Tutorial Files

The exercises in this tutorial depend on the performance task data sets that were created using the tutorial files in "Create the Champion Model Performance Data Sets" on page 122.

## Prepare to Use Dashboard Reports

The dashboard report directory is configured during the installation of SAS Model Manager. The default directory is **`\SASConfigDir\Lev#\AppData\SASModelManager3.1\Dashboard`**.

To configure a different directory to store the SAS Model Manager dashboard reports, follow these steps:

1. Connect to the SAS Workspace Server.
2. Create a new directory (for example, **`C:\Dashboard`**).

   *Note:* The directory must be located on a SAS Workspace Server or a network drive that is accessible by the SAS Workspace Server. Do not include special characters or spaces in the name of the directory.
3. Grant user permissions for the new directory. For example, perform the following tasks:
   - Grant Full Control permission to users who need to create subdirectories, write content, or delete content. This type of user includes a user who you will be adding (using SAS Management Console) to the Model Manager Administrator Users group or a user who is a SAS Administrator.
   - Grant Read, Write, and Execute permissions to users who need to create performance indicators and execute dashboard reports. This type of user includes a user who you will be adding (using SAS Management Console) to the Model Manager Advanced Users group.
   - Grant Read and Execute permissions to users who need only to view the dashboard reports. This type of user includes a user who you will add (using SAS Management Console) to the Model Manager Users group.

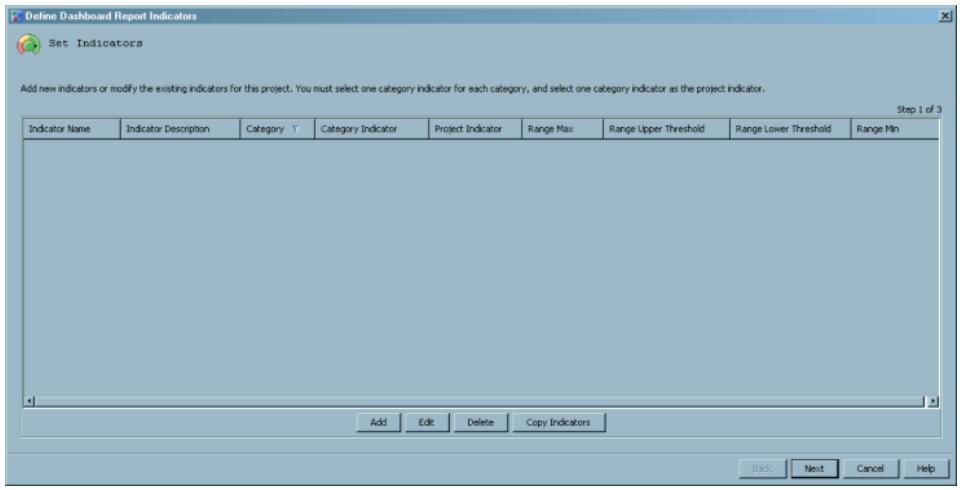
   *Note:* In a UNIX environment all SAS Model Manager users must be part of a group that has the appropriate group permissions. For more information, see "Creating Operating System Accounts in UNIX Environments" in Chapter 2 of *SAS Model Manager: Administrator's Guide* and "Configuring Users, Groups, and Roles" in Chapter 3 of *SAS Model Manager: Administrator's Guide*.
4. From SAS Management Console, expand the **Application Management** node on the **Plug-ins** tab.
5. Select and expand **Configuration Manager** ⇨ **SAS Application Infrastructure**.

6. Right-click **Model Manager JavaSvcs 3.1** and select **Properties**.
7. (optional) Click the **Settings** tab and then select **Model Manager Dashboard Options**. This setting enables you to modify formats, styles, and the indicator override. The indicator override configuration enables indicators that have conditions to be available when you add dashboard report indicators using the SAS Model Manager 3.1 Client.
8. Click the **Advanced** tab to modify the application dashboard directory. Change the property value for **App.DashboardDir** to the directory path that was configured.
9. Click **OK**.

### *Define Dashboard Report Indicators*

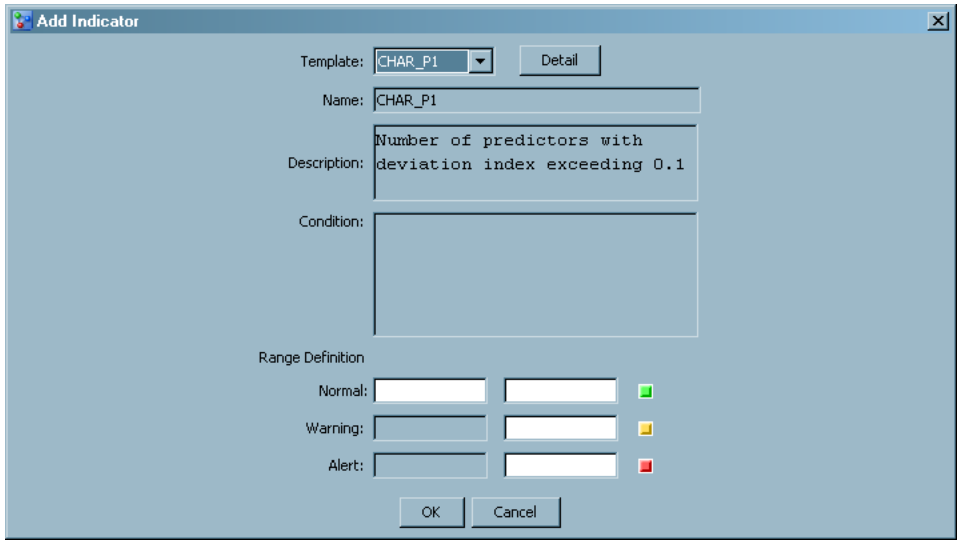
To define dashboard report indicators, follow these steps:

1. You must have at least one project that contains performance data before you continue to the next step. For more information, see "Create the Champion Model Performance Data Sets" on page 122.
2. Right-click the project folder in the Project Tree, and select **Define Dashboard Report Indicators** from the pop-up menu. The Define Dashboard Report Indicators window appears.




3. Click **Add**. The Add Indicator window appears.

   *Note:* If you want to copy indicators from an existing project, click **Copy Indicators** instead of **Add**, follow the prompts, and then skip to step 5.




a. Select a template from the **Template** drop-down list.

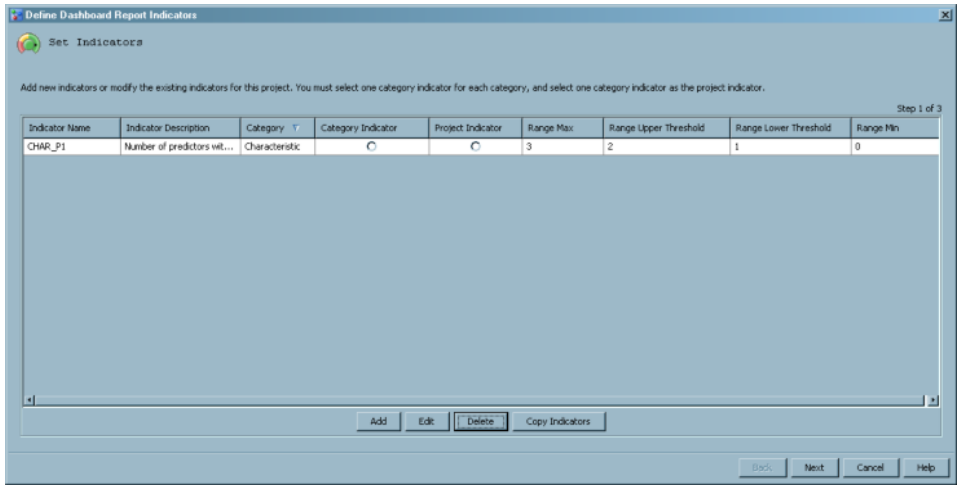
   *Note:* Click **Detail** to view information about the selected indicator template.

b. Enter values for the **Normal**, **Warning**, and **Alert** range definitions.

   ***Table 8.1*** *Example Performance Indicator Values*

| Indicator Name | Category | Normal | Warning | Alert |
|---|---|---|---|---|
| CHAR_P1 | Characteristic | 0 – 1 | 1 – 2 | 2 – 3 |
| GINIDECAY | Model Assessment | 0 – 0.2 | 0.2 – 0.4 | 0.4 – 0.6 |
| STAB_P1 | Stability | 0 – 1 | 1 – 2 | 2 – 3 |

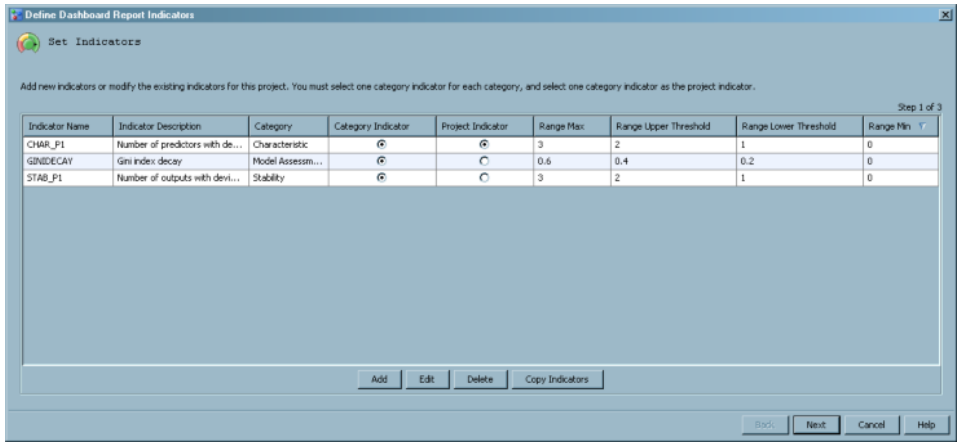
c. Click **OK**. The Define Dashboard Report Indicators window appears with information about the new indicator.



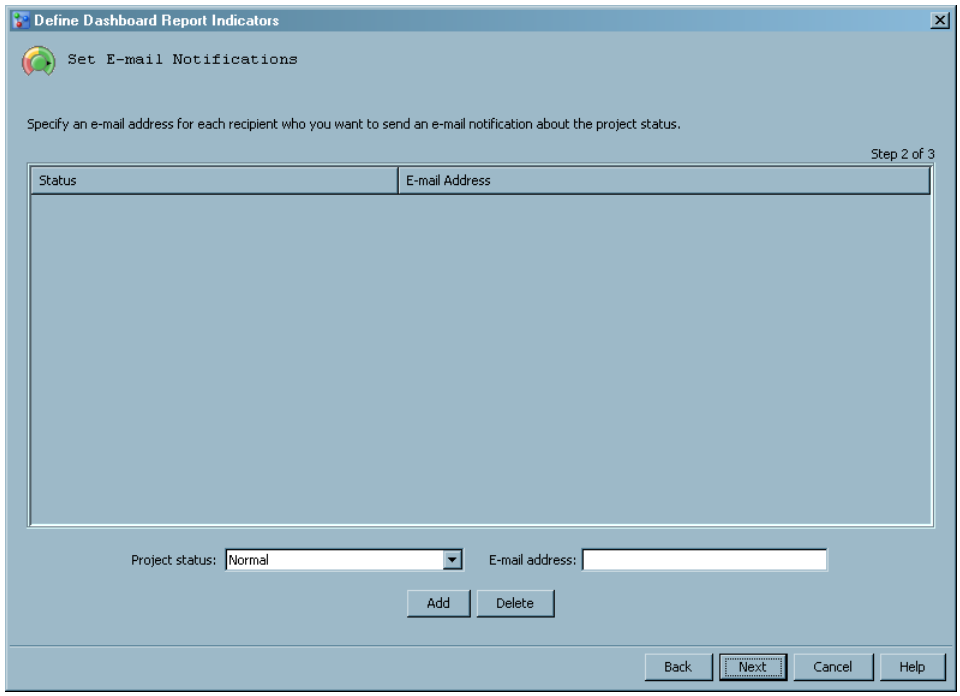

4. Repeat step 3 for each indicator that you want to add. To edit an existing indicator, select the indicator, and click **Edit**.

5. Select one **Category Indicator** for each category, and then select one indicator as the **Project Indicator**.

   *Note:* The indicator that you select as a project indicator must also be a category indicator.

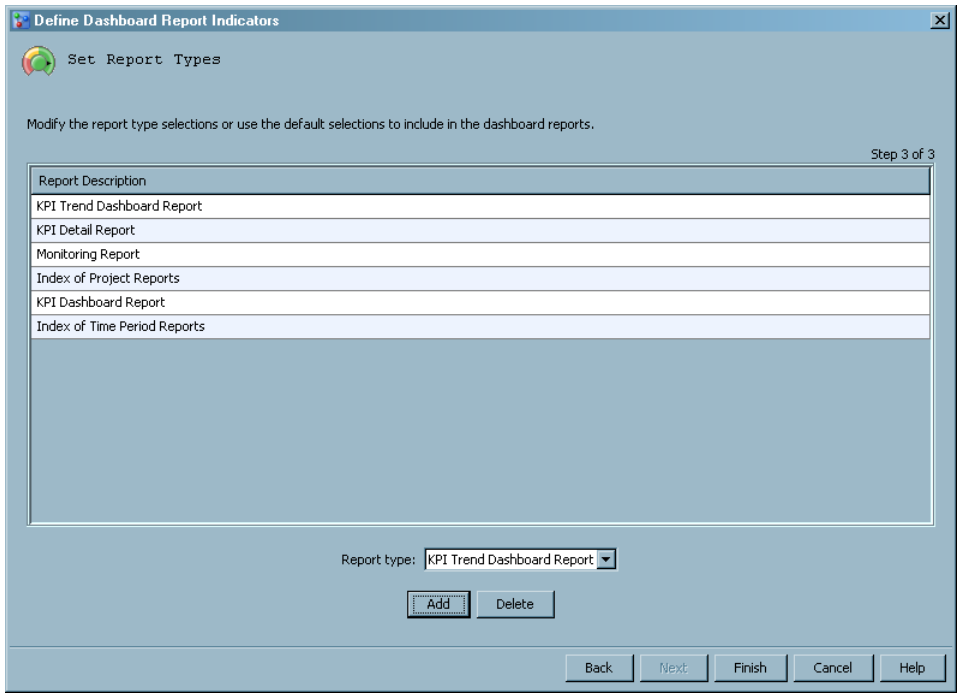



6. Click **Next**. The Define Dashboard Report Indicators window appears with information about setting up notifications.




7. Select a value from the **Project status** drop-down list, enter a value for **E-mail address**, and click **Add**. Repeat this step for each recipient who you want to send an e-mail notification to about a status. If you want an individual recipient to receive an e-mail notification for each status, you must repeat this step for each status. To delete an e-mail notification, select a project status, and click **Delete**.

8. Click **Next**. The Define Dashboard Report Indicators window appears with information about setting report types.

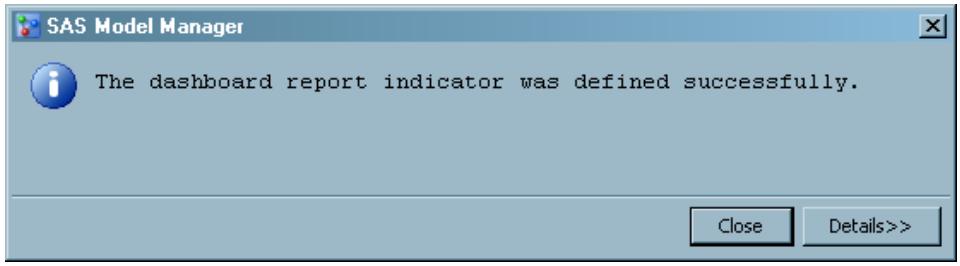



9. By default, all of the report types are selected. To change report types, follow these steps:

   a. To add a report type, select a value from the **Report type** drop-down list, and click **Add**.

   b. To delete a report type, select a value from the **Report Description** list, and click **Delete**.

10. Click **Finish**. An informational message is displayed indicating that the dashboard indicator was defined successfully.



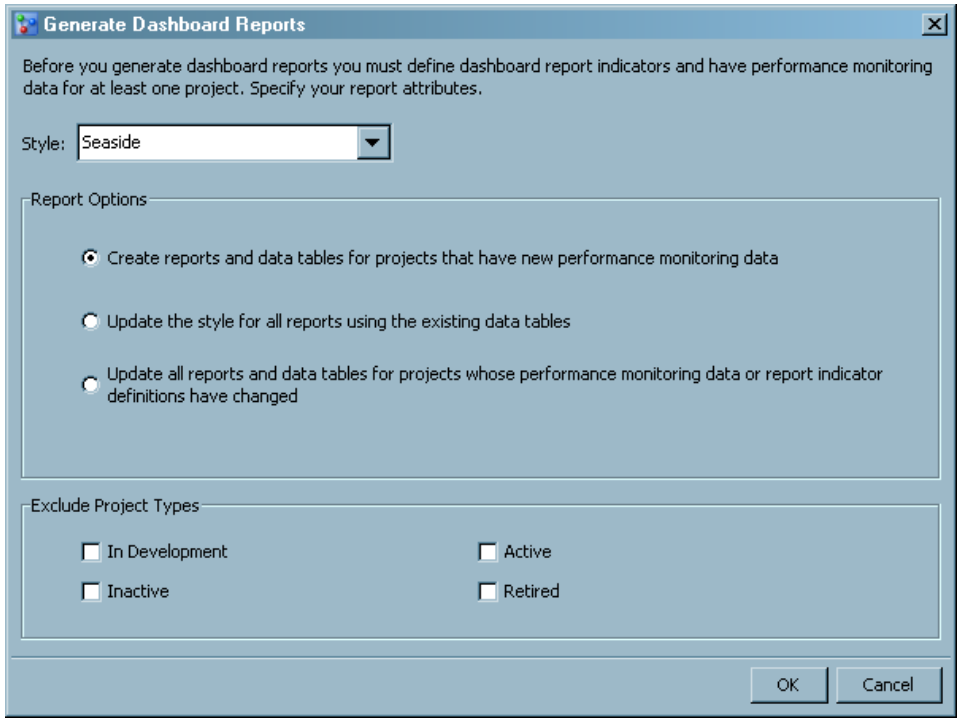

*Note:* You must define dashboard report indicators for all projects that you want to include in your dashboard reports.

## *Generate Dashboard Reports*

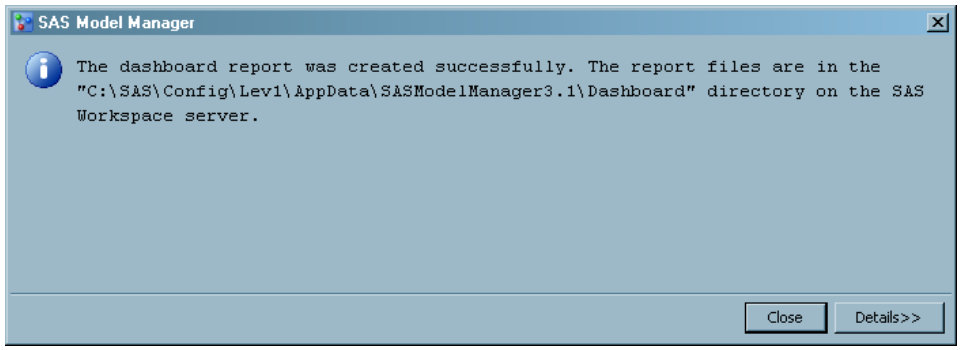
To generate the dashboard reports, follow these steps:

*Note:* Before you generate dashboard reports, you must have at least one project that contains performance data. That project must have at least one dashboard report indicator that has been defined.

1. Select **Tools** ⇨ **Generate Dashboard Reports** from the menu. The Generate Dashboard Reports window appears.




2. Select a style for the report from the **Select a style** drop-down list.
3. Select one of the following report options:
   - Create reports and data tables for projects that have new performance monitoring data.
   - Update the style for all reports, using the existing data tables.
   - Update all reports and data tables for projects whose performance monitoring data or report indicator definitions have changed.
4. (Optional) Select one or more project types that you want to exclude from the dashboard reports.
5. Click **OK**. You can view the progress of the dashboard reports in the status bar. A message appears that indicates whether the report was created successfully. The message also displays the location of the dashboard reports on the SAS Workspace Server. Here is an example: **`C:\SAS\Config\Lev1\AppData\SASModelManager3.1\Dashboard`**.



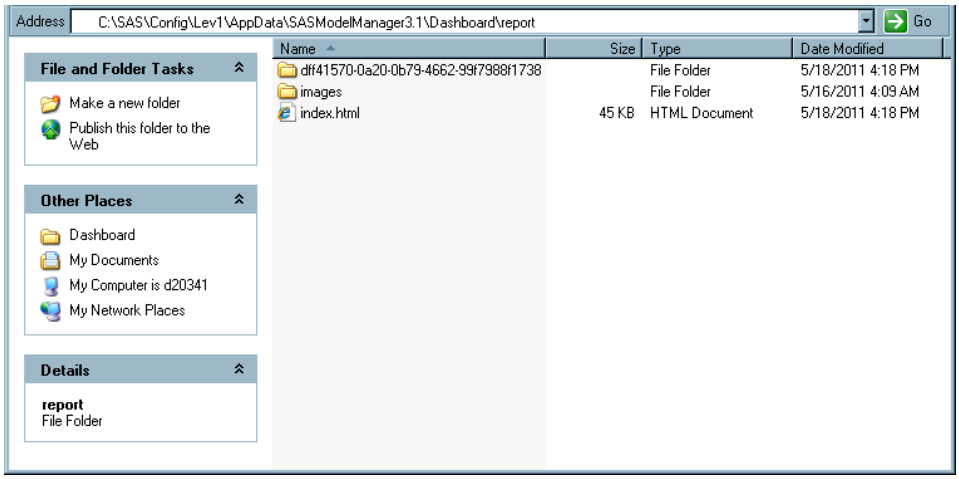

*Note:* SAS Model Manager administrators can configure a different location for the dashboard reports directory. If you set up notifications when you defined the dashboard indicators, the recipients will receive e-mail notifications for the configured statuses.

For more information about executing dashboard reports, see the *SAS Model Manager: User's Guide*.

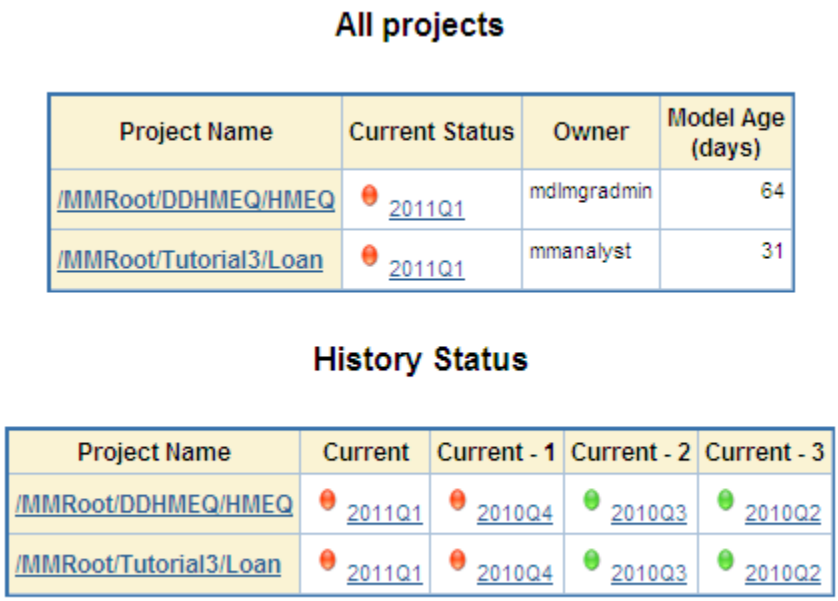
### *View the Dashboard Reports*

To view the dashboard reports, follow these steps:

1. Navigate to the **report** folder in the dashboard report directory on the SAS Workspace Server. Here is an example, **`C:\SAS\Config\Lev1\AppData\SASModelManager3.1\Dashboard`**.




2. Open the **`index.html`** file in your browser window. The SAS Model Manager Dashboard appears.




3. Select a project name or a status to view the associated dashboard reports. The Project Reports Index appears in a new window. If you select a status, only the dashboard reports for that time frame are displayed.

**Project Reports Index**

| Time | Status | Project Indicator | Report |
|---|---|---|---|
| 2011Q1 |  | Number of predictors with deviation index exceeding 0.1 | KPI Dashboard Report |
| | | | KPI Detail Report |
| | | | KPI Trend Dashboard Report |
| | | | Monitoring Report |
| 2010Q4 |  | Number of predictors with deviation index exceeding 0.1 | KPI Dashboard Report |
| | | | KPI Detail Report |
| | | | KPI Trend Dashboard Report |
| | | | Monitoring Report |
| 2010Q3 |  | Number of predictors with deviation index exceeding 0.1 | KPI Dashboard Report |
| | | | KPI Detail Report |
| | | | KPI Trend Dashboard Report |
| | | | Monitoring Report |
| 2010Q2 |  | Number of predictors with deviation index exceeding 0.1 | KPI Dashboard Report |
| | | | KPI Detail Report |
| | | | KPI Trend Dashboard Report |
| | | | Monitoring Report |

4. Select a report to view report details. The report details appear in the same window.

**KPI Detail Report**
**2011Q1**

| Category | Category Status | Category Indicator | Indicator | Indicator Status | Value |
|---|---|---|---|---|---|
| Characteristic |  |  | Number of predictors with deviation index exceeding 0.1 |  | 3.0000 |
| Model Assessment |  |  | Gini index decay |  | 0.4564 |
| Stability |  |  | Number of outputs with deviation index exceeding 0.1 |  | 0.0000 |

*Note:* To return to the Project Reports Index, select the browser's back button.

For more information about dashboard reports, see the *SAS Model Manager: User's Guide*.

*Chapter 9*

# Tutorial 8: Scoring a SAS Model Manager Model Using SAS Data Integration Studio



## Overview of Using Exported Models in SAS Data Integration Studio

The SAS Model Manager export feature enables you to register models and projects in SAS Metadata Repositories, making them available for other SAS products such as SAS Data Integration Studio and SAS Enterprise Guide. SAS Model Manager offers two export methods:

- Export a model
- Export a champion model

When you export a SAS Model Manager project champion model to the metadata repository, the result is a mining results object that contains the champion model of the project's default version. If the champion model in the project's default version is changed and the model is exported from the project level again to the same SAS metadata folder, the mining results object in the metadata repository is updated with the new champion model.

To illustrate an application that can use an exported SAS Model Manager project champion model, this tutorial uses SAS Data Integration Studio to connect metadata objects (including a mining results object) to create a scoring job. The tutorial also covers exporting the project champion model a second time with new content and updating the scoring job.

## Prerequisites

The exercises in this tutorial depend on some of the properties of the specific models that were added in Tutorial 3. Use the projects, versions, or models that are specified here. This tutorial is designed to follow Chapter 4, "Tutorial 3: Importing and Exporting Models," on page 69.

The scoring input and output tables from SMM31Tutorial.zip must be extracted and registered in SAS Management Console. If the data sets have not been extracted and registered, see "Prepare Tutorial 8 Data Sets and Models" on page 12 to extract and register the files.

This exercise requires SAS Data Integration Studio. Use the SAS Deployment Wizard to install the SAS Data Integration Studio client.

## Export a Project Champion Model from SAS Model Manager

In this exercise, you export a project champion model from SAS Model Manager in order for that model to be accessed and scored by SAS Data Integration Studio. When you export a project, you export the champion model from the project's default version.

*Note:* This task requires that you use a user ID that is a member of the SAS Model Manager Advanced User group or the SAS Model Manager Administrator group.

*Note:* If you create user-defined properties at the project level, these properties are exported with the champion model. User-defined properties might be helpful for scoring applications that search mining result objects for specific name value pairs. Each user-defined project property is stored in the SAS metadata repository as an Extension metadata object, which is a name-value pair.

1. Click the **Project perspective** button  . Expand **Tutorial3**.
2. Right-click **Loan** and select **Export Project Champion Model**. Click **Yes** when the message box prompts you about the unlocked project. The SAS Metadata Repository dialog box appears.
3. Double-click **Shared Data**. Double-click **Model Manager** and select **Tutorial3**. Click **OK**. An information message indicates whether the champion model was successfully exported. Click **Close** to close the message box.

## Score a Model Using a SAS Data Integration Studio Job

In this exercise, you create a SAS Data Integration Studio scoring job by using the Loan mining result from the SAS Metadata Repository. After you create the job, you run the job and view the output.

### *Open the SAS Data Integration Studio Desktop*

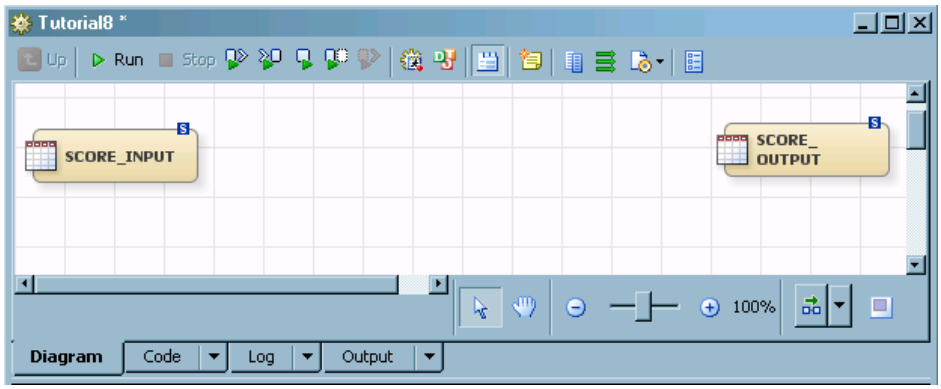
To log on to SAS Data Integration Studio:

1. Launch SAS Data Integration Studio.
2. If prompted, create a SAS Metadata Profile for the SAS Metadata server.
3. Log on with the profile for SAS Metadata server.

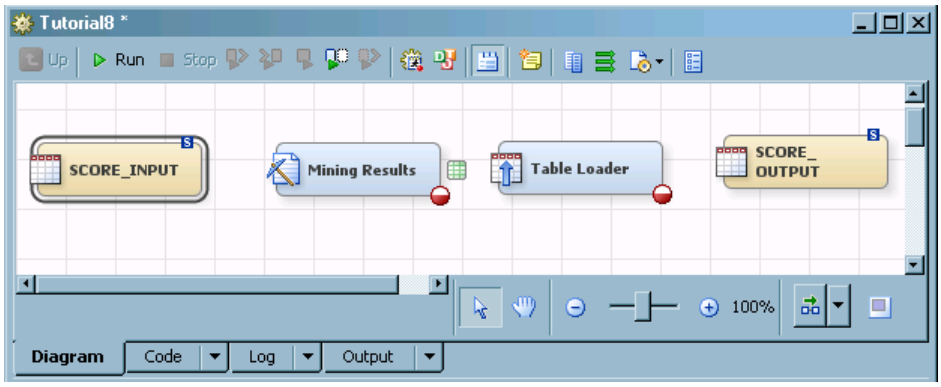
### *Create a New Job*

To create a new job, follow these steps:

1. Use the New Job Wizard to add the job:
   a. From the SAS Data Integration Studio window, select **My Folder**. Then select **New** ⇨ **Job**. The New Job dialog box appears.
   b. In the **Name** box, enter `Tutorial8` and click **OK**.
   c. Click the **Inventory** tab, expand **Table**, and find the tables **SCORE_INPUT** and **SCORE_OUTPUT**.
   d. Click and drag **SCORE_INPUT** to the **Diagram** tab. Click and drag **SCORE_OUTPUT** to the **Diagram** tab. Position the **SCORE_INPUT** node farthest to the left. Position the **SCORE_OUTPUT** node farthest to the right. These nodes are the beginning and ending nodes in the diagram. Leave enough space between them for two additional diagram nodes to occupy.




   e. Click the **Transformations** tab and expand **Access**. Select and drag **Table Loader** to the **Diagram** tab. Place **Table Loader** node before the **SCORE_OUTPUT** node.
   f. From the **Transformations** tab, expand **Data**. Select and drag **Mining Results** to the **Diagram** tab. Place the **Mining Results** node between the **SCORE_INPUT** node and the **Table Loader** node. Here is the **Diagram** tab:



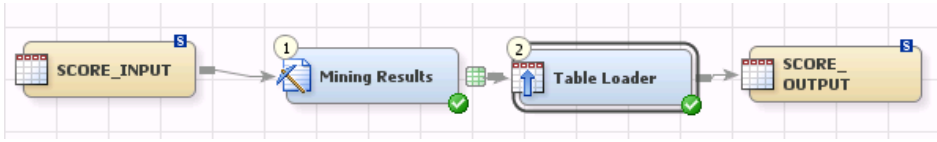

g. Double-click the **Mining Results** node. The Mining Results Properties window appears. Click the **Mining Results** tab, expand **Mining results**, and select **Loan**. The UUID in the **Key** box is the UUID of the **Loan** project in SAS Model Manager.

2. Click the **Target Table Columns** tab. Expand **OutputTable**, select **score**, and click . Click **OK**.

3. Drag the output handle from the **SCORE_INPUT** node to the **Mining Results** node. The half-filled circle on the **Mining Result** node is changed to a check mark to indicate that the node requirements have been met.

4. Drag the output handle from the **Mining Results** node to the **Table Loader** node.

5. Drag the output handle from the **Table Loader** node to the **SCORE_OUTPUT** node. The half-filled circle on the **Table Loader** node is changed to a check mark to indicate that the node requirements have been met. Here is the diagram:

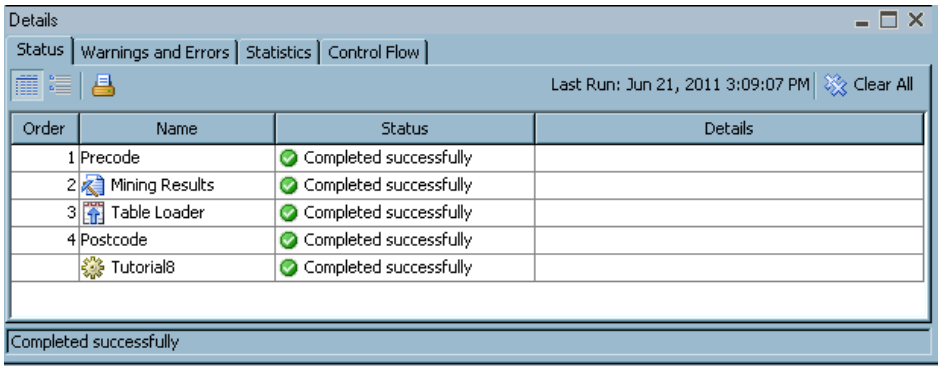



6. Save the job. Click **File** ⇨ **Save** .

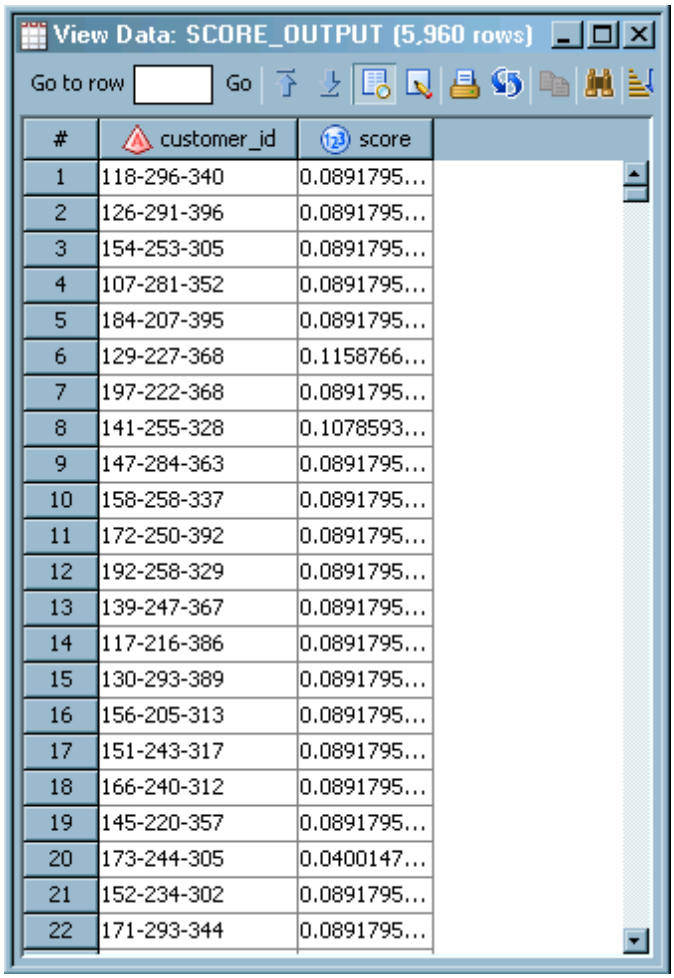
## *Run the SAS Data Integration Studio Scoring Job*

To run the job and view the output, follow these steps:

1. On the **Diagram** tab, select Run. The Tutorial8 job runs. Here is the job status:




2. To view the output, right-click the **SCORE_OUTPUT** node and select **Open**. Here is the output:



View Data: SCORE_OUTPUT (5,960 rows)

Go to row Go

| # | customer_id | score |
|---|---|---|
| 1 | 118-296-340 | 0.0891795... |
| 2 | 126-291-396 | 0.0891795... |
| 3 | 154-253-305 | 0.0891795... |
| 4 | 107-281-352 | 0.0891795... |
| 5 | 184-207-395 | 0.0891795... |
| 6 | 129-227-368 | 0.1158766... |
| 7 | 197-222-368 | 0.0891795... |
| 8 | 141-255-328 | 0.1078593... |
| 9 | 147-284-363 | 0.0891795... |
| 10 | 158-258-337 | 0.0891795... |
| 11 | 172-250-392 | 0.0891795... |
| 12 | 192-258-329 | 0.0891795... |
| 13 | 139-247-367 | 0.0891795... |
| 14 | 117-216-386 | 0.0891795... |
| 15 | 130-293-389 | 0.0891795... |
| 16 | 156-205-313 | 0.0891795... |
| 17 | 151-243-317 | 0.0891795... |
| 18 | 166-240-312 | 0.0891795... |
| 19 | 145-220-357 | 0.0891795... |
| 20 | 173-244-305 | 0.0400147... |
| 21 | 152-234-302 | 0.0891795... |
| 22 | 171-293-344 | 0.0891795... |

### *Verify the Model Code Used in the Job*

To verify that you have used the correct model, view the model code that was used in the SAS Data Integration Studio job.

Click the **Code** tab and scroll down through the lines until you find the following comment block:.

```
   *-------------------------------------------------------------*;
* TOOL: Score Node;
* TYPE: ASSESS;
* NODE: Score;
*-------------------------------------------------------------*;
*-------------------------------------------------------------*;
* EM SCORE CODE;
* VERSION: 7.1;
* GENERATED BY: mdlmgradmin;
* CREATED: 19JAN2011:14:19:11;
*-------------------------------------------------------------*;
*-------------------------------------------------------------*;
* TOOL: Input Data Source;
* TYPE: SAMPLE;
* NODE: Ids;
*-------------------------------------------------------------*;
```

```
*-------------------------------------------------------------*;
* TOOL: Regression;
* TYPE: MODEL;
* NODE: Reg;
*-------------------------------------------------------------*;
************************************;
*** begin scoring code for regression;
************************************;
```

The **NODE** value that is associated with **TYPE: MODEL** is the model name. In this case, the model name is **Reg**.

## Declare and Export a New Champion Model in SAS Model Manager

In this exercise, you declare a different model as the champion model after an initial project champion model has been exported to the SAS Metadata Repository. You then export the new project champion model to the metadata repository.

To declare and export a new champion model, do the following:

1. In the SAS Model Manager Project Tree, expand the **Tutorial3** folder, the **Loan** project, the **2011** version, and the **Models** folder.
2. Right-click the **Tree1** model and select **Set Champion Model**.
3. When prompted to confirm the change, click **Yes**.
4. Right-click the **Loan** project and select **Export Project Champion Model**. Click **Yes** in the message window. The SAS Metadata Repository window appears.
5. Double-click **Shared Data**, double-click **Model Manager**, select **Tutorial3**, and click **OK**. Click **Yes** in the confirmation message box to replace the champion model.

   An information message indicates whether the champion model was successfully exported. Click **Close** to close the message box.

For more information about this task, see the *SAS Model Manager 3.1: User's Guide*.

## Update the Job to Use the Latest Champion Model

This exercise demonstrates the steps to update the SAS Data Integration Studio job after you change the champion model in SAS Model Manager. After you export the **Loan** project from SAS Model Manager, SAS Data Integration Studio recognizes a new mining results object.

To update the job, follow these steps:

1. Close and reopen Tutorial8.
   a. Click the Tutorial8 window and select **File** ⇨ **Close**.

b. Click the **Folders** tab and expand **My Folder**. Double-click **Tutorial8** to reopen the job. When the job reopens, it uses the new score code from the Mining Result object and regenerates the code that is associated with the job.

2. In the **Tutorial8** diagram, right-click the **Mining Results** node and select **Properties**. The Mining Results Properties window appears. Click the **Mining Results** tab. The **Loan** mining result is highlighted. The **Algorithm** box shows that the model is a **DecisionTree** model.

3. Click the **Model Attributes** tab. A message box might appear that warns of potential table changes if you change the mining result. Click **Yes**.

4. Click **View Source Code**. Scroll to the top of the window. Compare the text in the comment tags to the Tree 1 model code in SAS Model Manager. They are the same, as shown in the displays below.

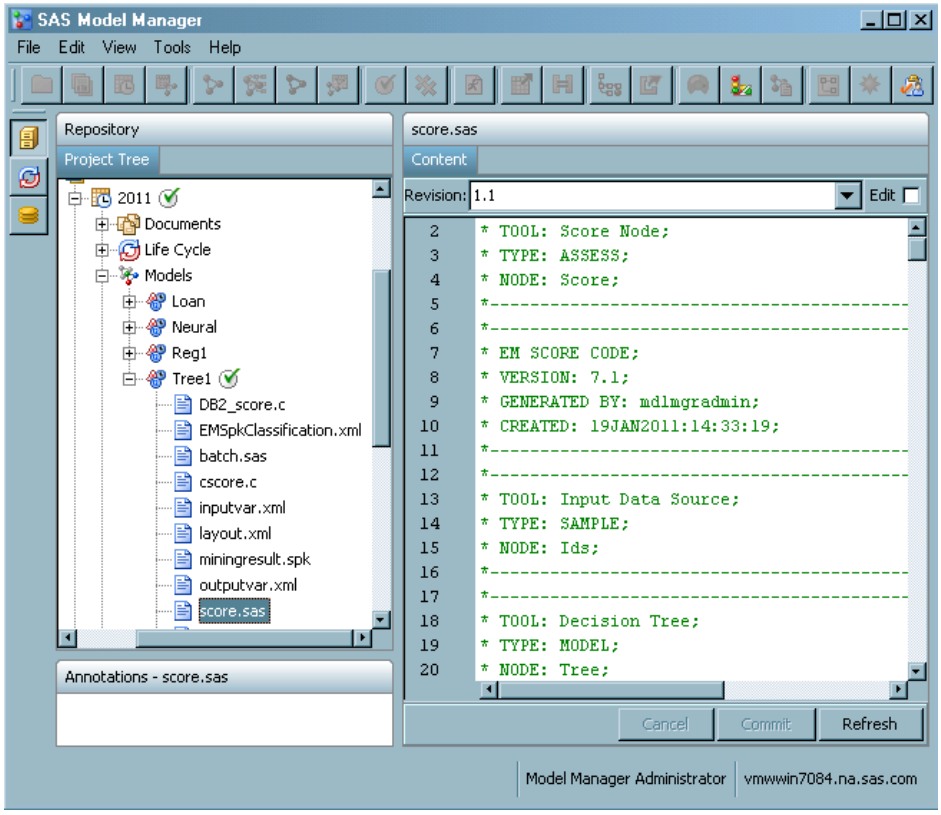
   To view the model code in SAS Model Manager, follow these steps:

   a. Log on to SAS Model Manager and expand the following Project Tree nodes:

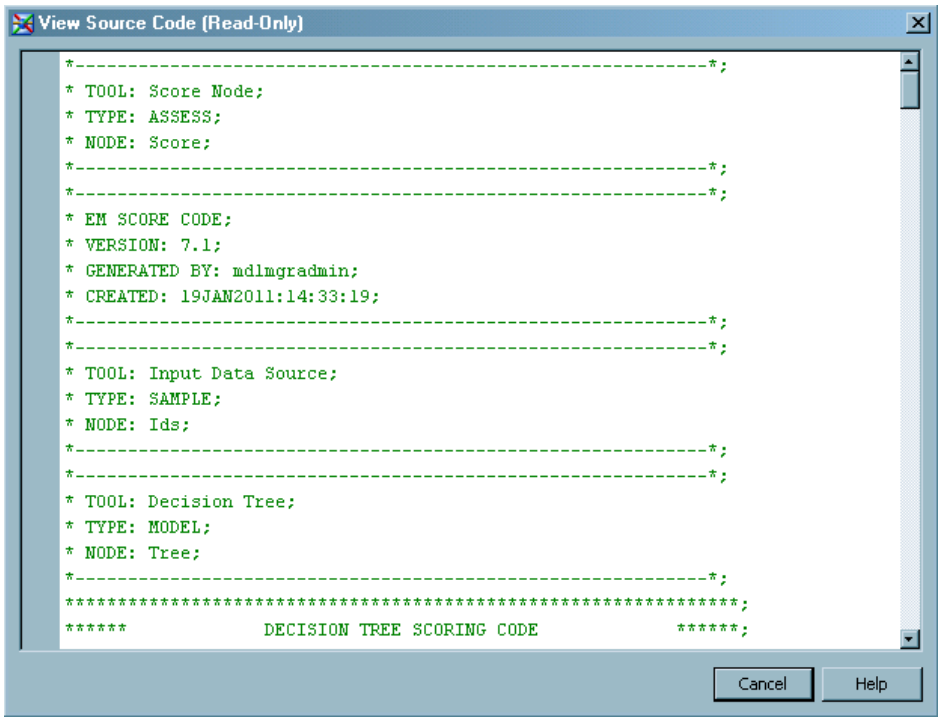
      - **Tutorial3** folder
      - **Loan** project
      - **2011** version
      - **Models** folder
      - **Tree 1** model

   b. In the **Tree 1** model, select **score.sas**. The model code appears in the **Content** view.

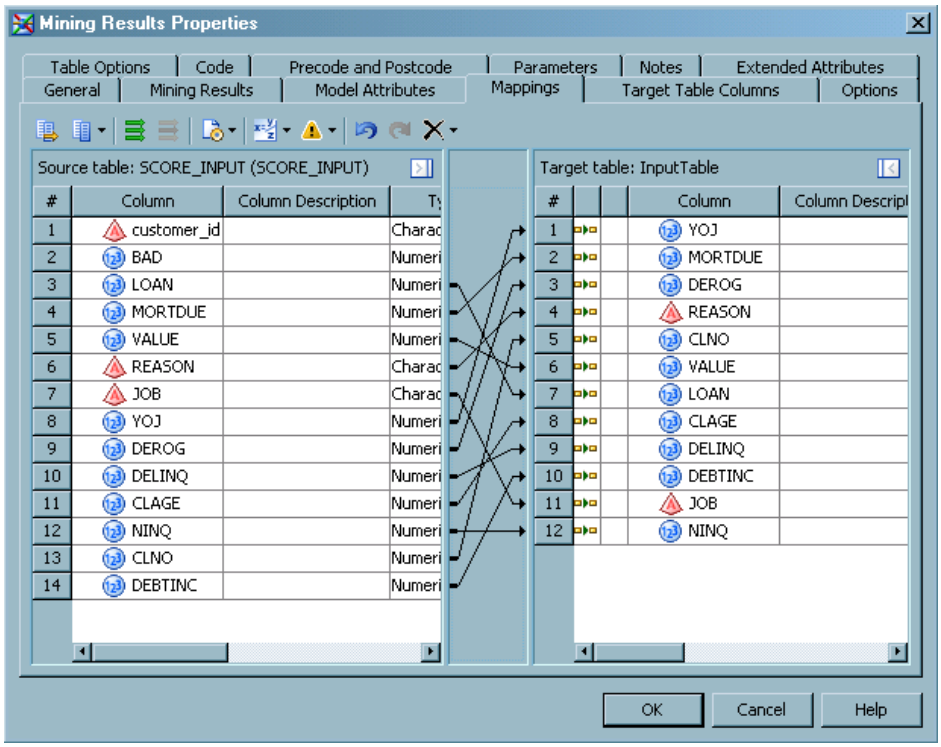
   Here is the Tree 1 score code in SAS Model Manager.




Here is the Tree 1 score code in SAS Data Integration Studio.

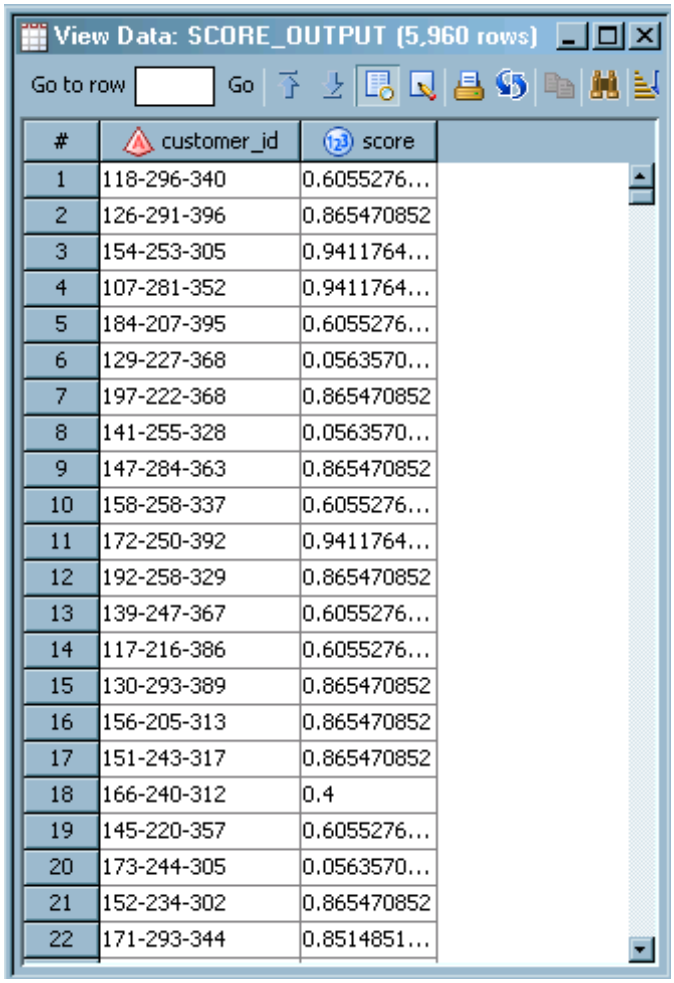



5. Click **Cancel** to close the View Source Code window.
6. Identify the variables to be used in the transform output. Click the **Target Table Columns** tab. If **score** and **customer_id** are not in the **Selected** list, follow these steps:
   a. From the **Available** list, select **score** and click ▶ to move the **score** variable to the **Selected** list.
   b. From the **Available** list, expand **SCORE_INPUT**, select **customer_id**, and click ▶ to move the **customer_id** variable to the **Selected** list.
7. Click the **Mappings** tab. Right-click the space between the two lists of variables and select **Map All**. Here is the **Mappings** tab:




Click **OK**.

8. Double-click the **Table Loader** node and click the **Mappings** tab.
9. Right-click the space between the two lists of variables and select **Map All**. Click **OK**.
10. On the **Diagram** tab, select Run. The **Tutorial8** job runs.
11. To view the output, right-click the SCORE_OUTPUT node and select **Open**. Here is a partial view of the output:



View Data: SCORE_OUTPUT (5,960 rows)

Go to row Go

| # | customer_id | score |
|---|---|---|
| 1 | 118-296-340 | 0.6055276... |
| 2 | 126-291-396 | 0.865470852 |
| 3 | 154-253-305 | 0.9411764... |
| 4 | 107-281-352 | 0.9411764... |
| 5 | 184-207-395 | 0.6055276... |
| 6 | 129-227-368 | 0.0563570... |
| 7 | 197-222-368 | 0.865470852 |
| 8 | 141-255-328 | 0.0563570... |
| 9 | 147-284-363 | 0.865470852 |
| 10 | 158-258-337 | 0.6055276... |
| 11 | 172-250-392 | 0.9411764... |
| 12 | 192-258-329 | 0.865470852 |
| 13 | 139-247-367 | 0.6055276... |
| 14 | 117-216-386 | 0.6055276... |
| 15 | 130-293-389 | 0.865470852 |
| 16 | 156-205-313 | 0.865470852 |
| 17 | 151-243-317 | 0.865470852 |
| 18 | 166-240-312 | 0.4 |
| 19 | 145-220-357 | 0.6055276... |
| 20 | 173-244-305 | 0.0563570... |
| 21 | 152-234-302 | 0.865470852 |
| 22 | 171-293-344 | 0.8514851... |

*Chapter 10*

# Tutorial 9: Retraining Models



## Overview of Retraining Models

Using SAS Model Manager, you can retrain models to respond to data and market changes. Retraining models enables you to update models and to improve model performance. When you define a model retrain task, you can select multiple models to be retrained at the same time. The definition of the model retrain task includes the destination version and training data source. The destination version is an existing version or new version that is associated with the selected project and that stores the retrained model information. The training data source contains new data for retraining the selected models.

The model retrain task enables you to specify a location to store comparison reports and retrain results. When you select the models to include in the comparison report, you can use the training data source or select a different data source to compare the performance of the new models. The model retrain task also enables you to specify the report options, including the name, format, and style of the comparison report. When you define e-mail notifications for a model retrain task, they are sent after you execute a model retrain task.

By default, the champion model in the default version for the selected project is retrained if a model is not selected. After you execute a model retrain task, if the **Register new trained model** option was selected, SAS Model Manager registers the new models to the destination version. The comparison report is stored in the **Model Retrain** folder, as well as in the report folder on the SAS Workspace Server that was specified when the model retrain task was defined.

*Note:* Only models that are created by using SAS Enterprise Miner or R models can be retrained.

In this tutorial, you perform the following tasks:

- define a model retrain task

- execute the model retrain task
- view the new retrained models and comparison report

## Prerequisites

The exercises in this tutorial depend on some of the properties of the specific project, version, and models that were added in Chapter 4, "Tutorial 3: Importing and Exporting Models," on page 69 and Chapter 8, "Tutorial 7: Creating Performance Monitoring Reports and Using Dashboard Reports," on page 121. The folder path in the Project Tree is **MMRoot** ⇨ **Tutorial3** ⇨ **Loan** ⇨ **2011**.

Before you execute the Define a Model Retrain Task, complete the following tasks:

- If you want to retrain the project champion model, ensure that the champion model in the default version is set.
- Verify that the training data set that is used in Tutorial 3 has been registered in the SAS Metadata Repository using SAS Management Console, so that you use the same data set as the training data source.
- Verify that the appropriate project and model properties are set.

  Here is a list of properties:

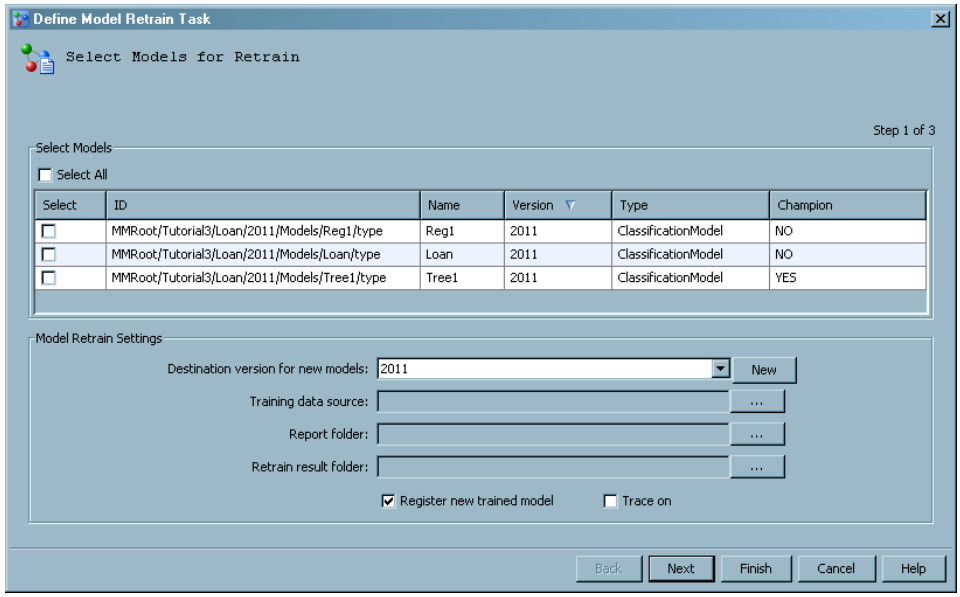
  Project Properties
  - Training Target Variable: `bad`
  - Target Event Value: `1`
  - Class Target Level: `Binary`
  - Output Event Probability Variable: `score`

  Model Properties
  - Score Code Type: `Data Step`
- Verify that all of the project output variables are mapped to the corresponding model output variables.

## Define a Model Retrain Task

To define a model retrain task, follow these steps:

1. Right-click the **Loan** project name and select **Define Model Retrain Task**. The Define Model Retrain Task wizard appears.




2. Select the **Reg1** model to be retrained.

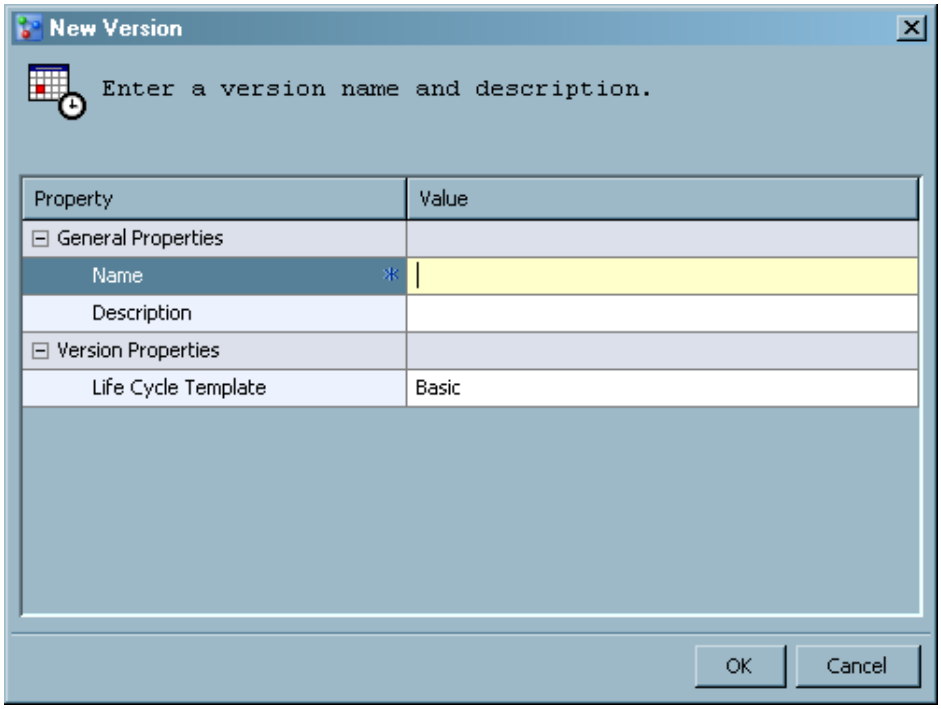
   *Note:* To select all models, you can select the **Select All** check box. If you do not select a model, the champion model in the default version for the selected project is retrained.

3. Select **2011** as the destination version for new models.

   *Note:* If you do not select a destination version, the default location is used for the destination of the new retrained models.

   (Optional) To create a new version to store the retrained models, follow these steps:

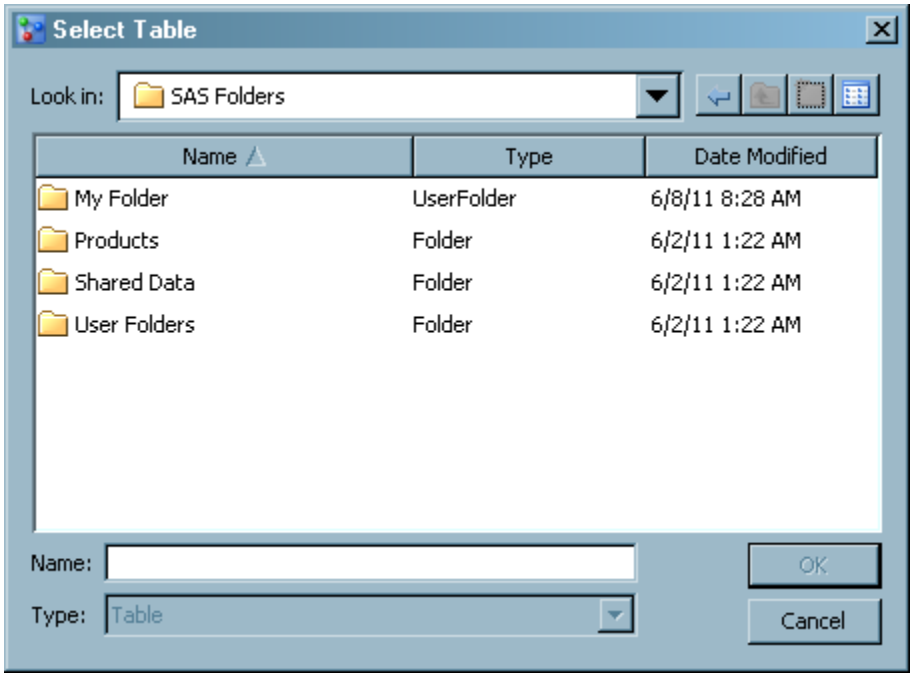
   a. From the Select Models for Retrain page of the wizard, click **New**. The New Version window appears.



b. Enter a name of the new version and select a life cycle template. Entering a description of the new version is optional.

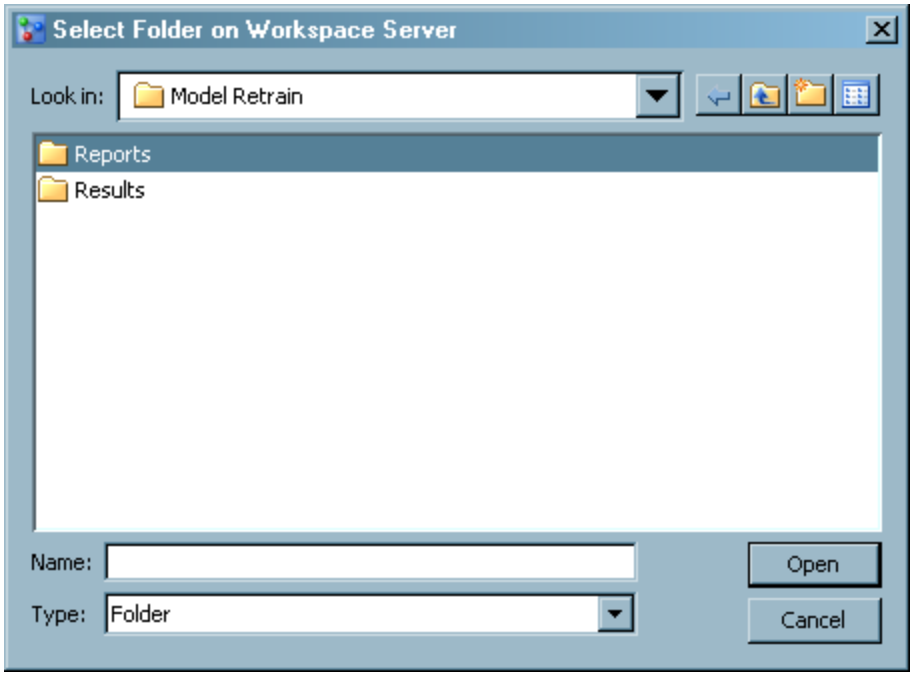
c. Click **OK**. You are then returned to the Define Model Retrain Task wizard.

4. Click … for the **Training data source** field



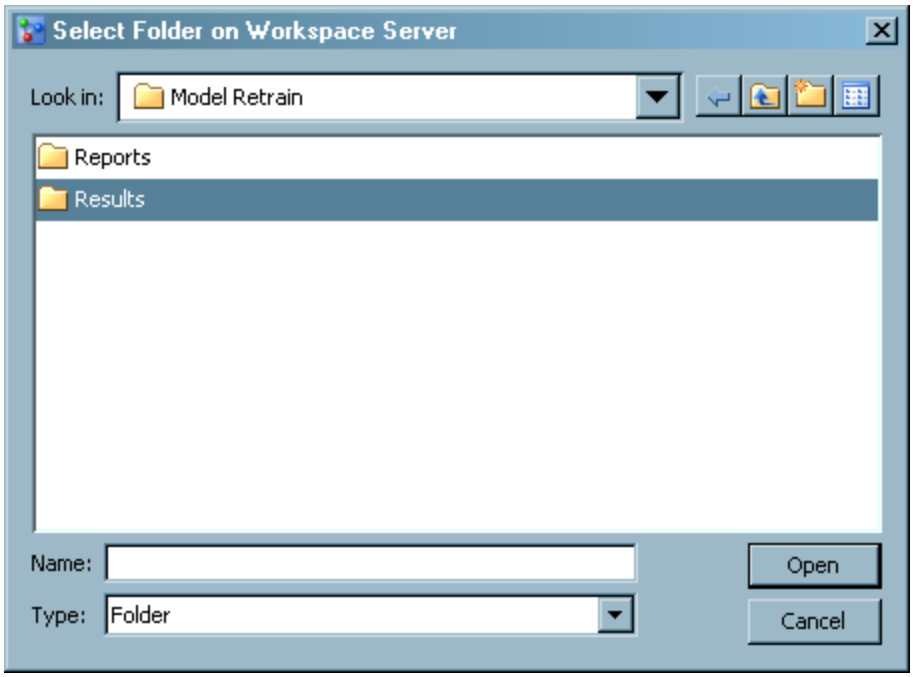

Select the HMEQ_TRAIN data set that is located in the **`\Shared Data\Model Manager\Tutorial3\`** directory of the SAS Metadata Repository. Click **OK**. You are then returned to the Define Model Retrain Task wizard.

5. Click … to select a value for the **Report folder**. This value is the location of where to store the comparison report, in addition to storing the report in the **Model Retrain** folder. By default, the report is stored in the SAS session's working folder on the SAS Workspace Server. You can also create subfolders in which to store the report. Here is an example: **`\\myserver.com\c:\Users\mmanalyst\Documents\My SAS Files\9.3\Model Retrain\Reports`**

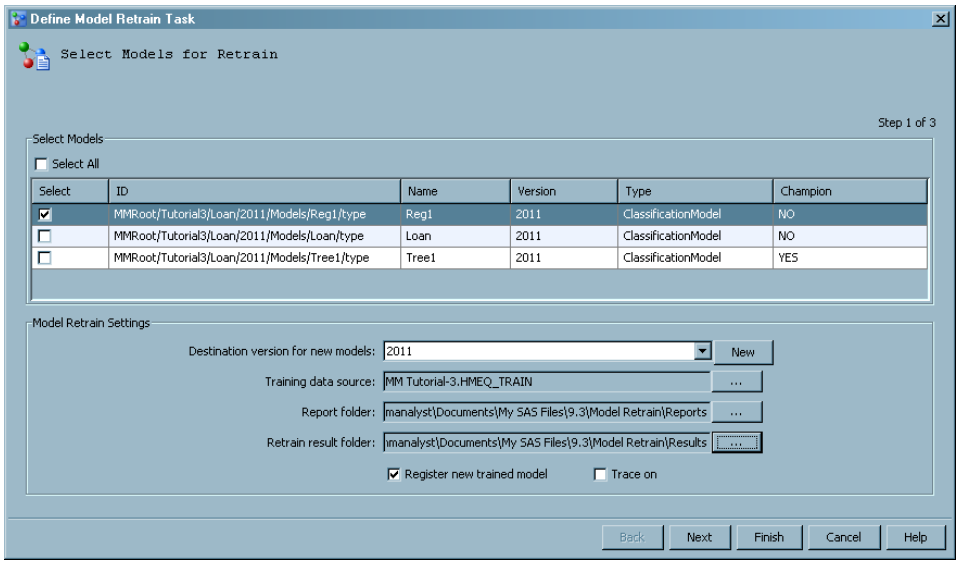



6. Click … to select a value for the **Results folder**. This value is the location of the retrain results folder to store the model training results.

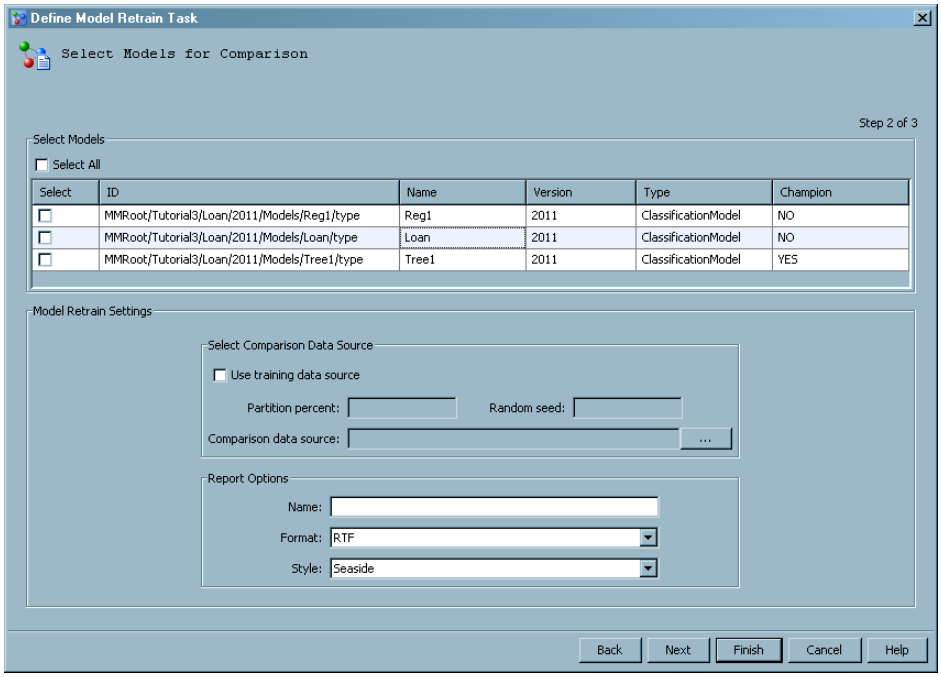
   *Note:* This setting is for informational purposes only. The data sets and files that are created during model retraining are stored in this location. By default, the training results are stored in the SAS session's working folder on the SAS Workspace Server. You can also create subfolders in which to store the results. Here is an example: **`\\myserver.com\c:\Users\mmanalyst\Documents\My SAS Files\9.3\Model Retrain\Results`**




7. Select **Register new trained model** to register the new models. If this option is not selected, the new models are not registered in the destination version in the Project Tree, and they are not saved to the model repository on the SAS Content Server.




8. (Optional) Select **Trace On** to print trace information to the SAS log file.
9. Click **Next**. The **Select Models for Comparison** page appears.




10. Select the **Loan** model to compare it to the retrained model.

    *Note:* To select all models, you can click **Select All**. If you do not select a model, the champion model in the default version for the project is used to perform the comparison.

11. Select a comparison data source. Take one of the following steps:

    - Select **Use training data source** if you want to use **`HMEQ_TRAIN`** as the comparison data source. For this example the default values are used. However, you can either use the whole training data source to compare or you can partition it into two parts, based on partition percent and random seed. The percentage that is specified is the percentage of data that is used for model comparison; the other part of the data is used for training. The random seed value is used to generate the training data, based on the random sampling method.

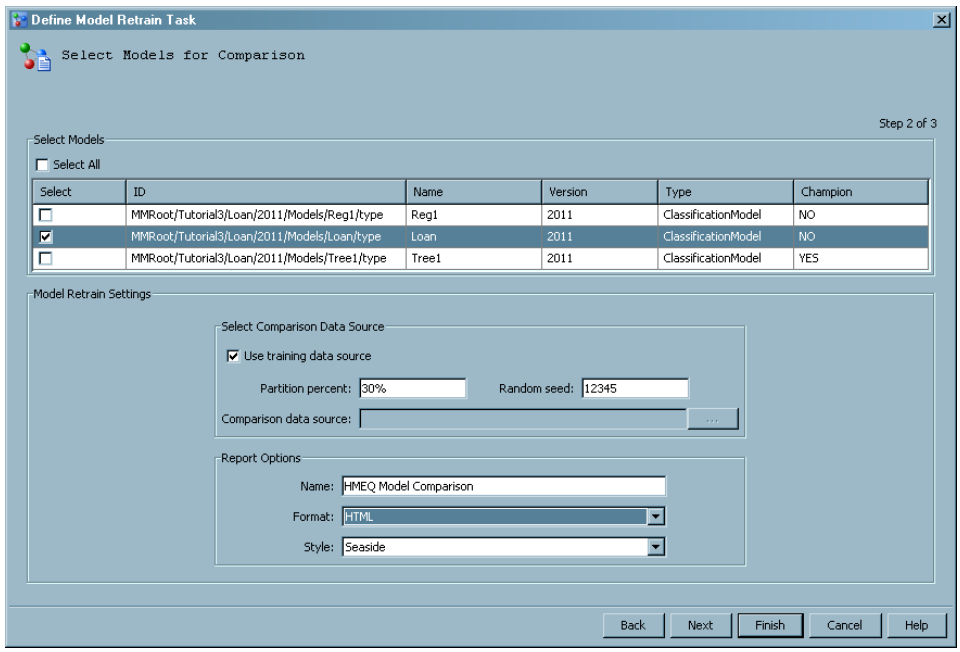
- Click ... for the **Comparison data source** field to select a performance data set.

12. To specify the report options, follow these steps.

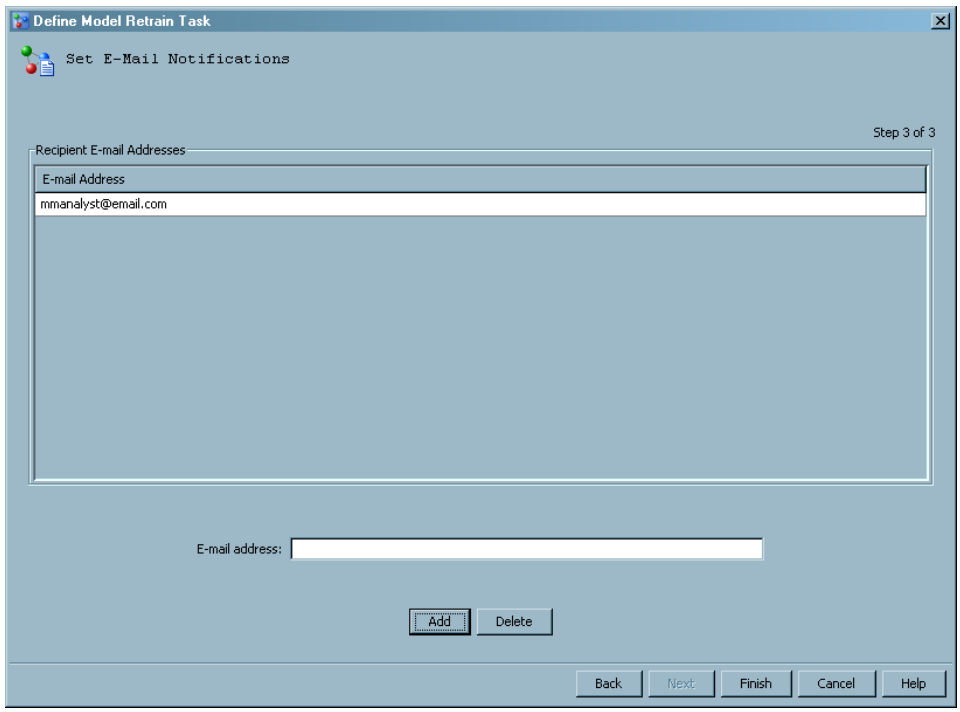
    a. Enter a report name. Here is an example: **`HMEQ Model Comparison`**.

    b. Select the **`HTML`** format for the report output. The default is **`RTF`**. The other available formats are **`PDF`**, **`HTML`**, and **`Excel`**.

    c. Select a style for the report. The default selection is **`Seaside`**. The other available styles are **`Meadow`**, **`Harvest`**, and **`Default`**.




13. Click **Next**. The **Set E-Mail Notifications** page appears.



14. (Optional) To send the training results by e-mail, enter an e-mail address or multiple e-mail addresses that are separated by a comma or blank, and then click **Add**. To delete a recipient, select the recipient's e-mail address and click **Delete**.
15. Click **Finish**. The SAS code is generated and placed in the **Model Retrain** folder of the associated project.

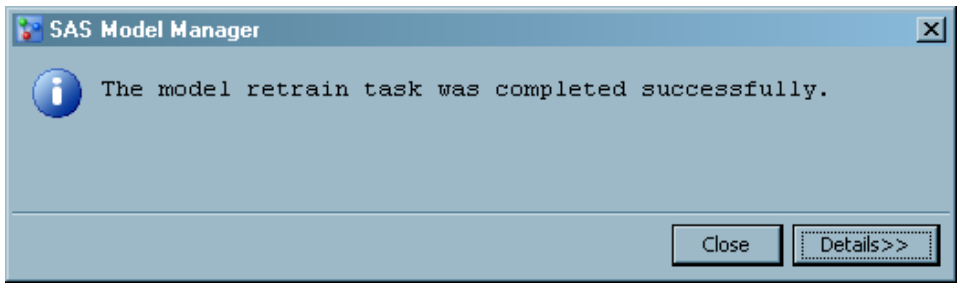
## Execute a Model Retrain Task

The prerequisites for retraining a model must be completed and a model retrain task must be defined before you can execute a model retrain task.

To execute a model retrain task, follow these steps:

1. Expand the project folder.
2. Right-click the **Model Retrain** folder, and then select **Execute** from the pop-up menu.

   *Note:* The model retrain task is executed as a background process. You can view the progress of the model retrain task in the status bar at the bottom of the SAS Model Manager application window.
3. When the model retrain task has finished executing, a message appears. Click **Close**.




*Note:* In the previous exercise, you chose to register the retrained model. The retrained model is now available in the **Models** folder of the selected destination version in the Project Tree. If the model retrain task does not execute successfully, click **Details**, or look for error messages in the SAS log (**`ModelRetrain.log`**) that is located on the SAS Content Server. You can find the SAS log and the retrained model comparison report in the **Model Retrain** folder.

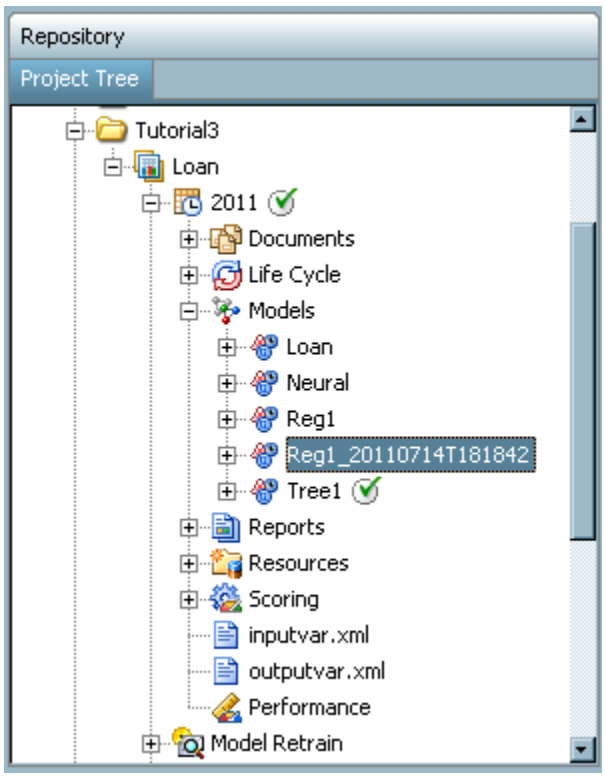
## Viewing Retrained Models and Model Comparison Reports

After a model retrain task is executed, the new retrained models are available in the **Models** folder within the destination version. In addition, the retrained model comparison report is available in the **Model Retrain** folder for the associated project.

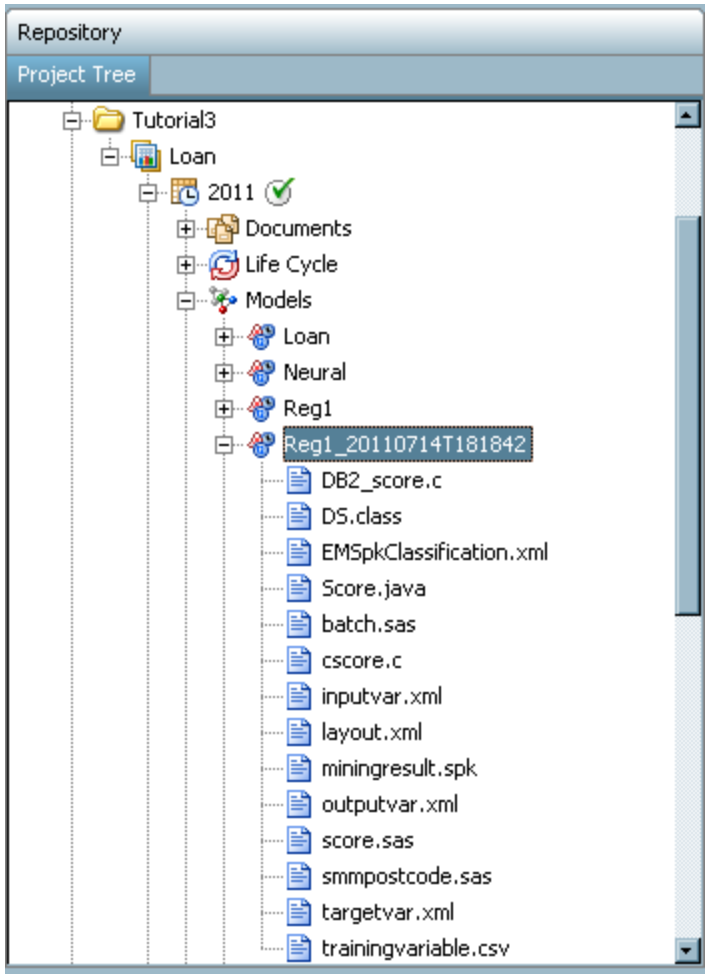
### *View Retrained Models*

To view retrained models, follow these steps:

1. Expand the destination version node **2011** to see the new retrained model in the **Models** folder.




2. Expand the new retrained model folder to view its contents.

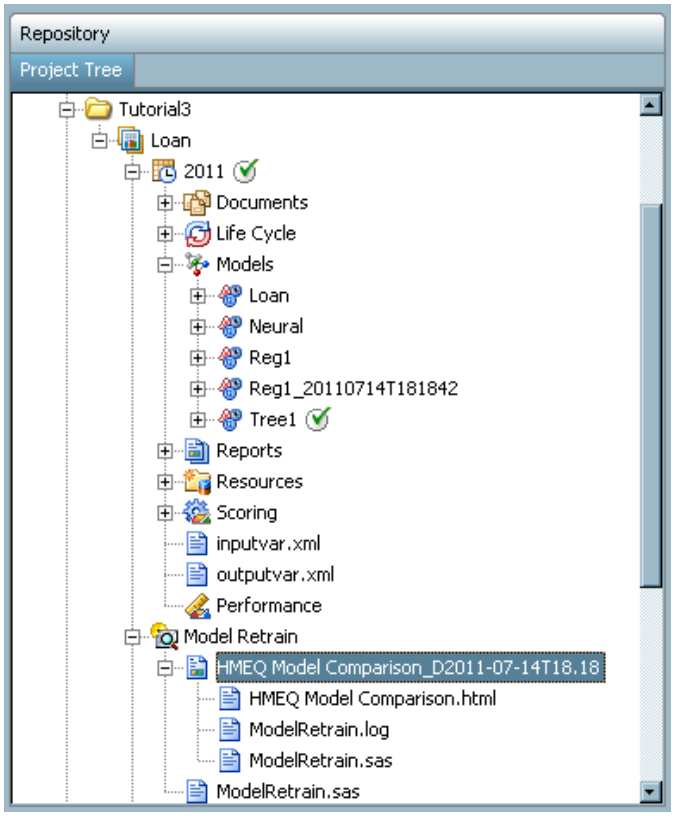



### *View Model Comparison Reports for Retrained Models*

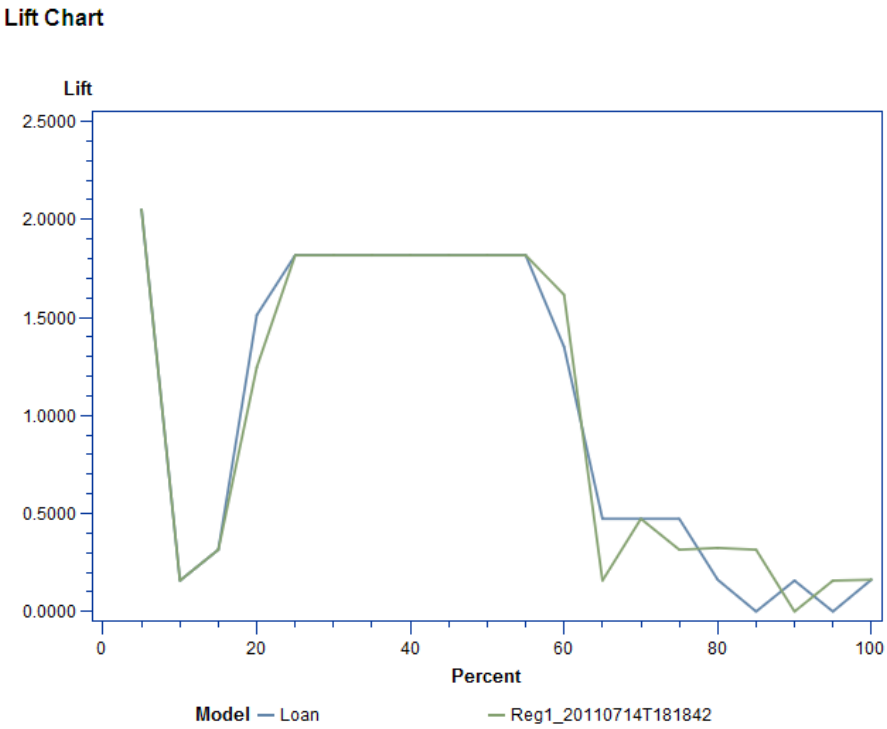
To view a model comparison report, follow these steps:

1. Expand the report folder that you specified when you defined the model retrain task (for example, *HMEQ Model Comparison*`_DYYYY-MM-DDTMM:SS`). The report folder is located in the **Model Retrain** folder for the associated project.

   *Note:* The name of the report folder also contains a timestamp in the format of `_DYYYY-MM-DDTMM:SS` that is supplied by the system when the report is created.




2. Right-click the report output file, and select **Open** from the pop-up menu. Specify user credentials when you are prompted. The report appears in your browser window. Here is an example of a Lift Chart that is part of the model comparison report.




*Note:*

You can also view the model retrain report in the following ways:

- Navigate to the report folder location on the SAS Workspace Server that you specified when defining the model retrain task. Here is an example: **\ \myserver.com\c:\Users\mmanalyst\Documents\My SAS Files \9.3\Model Retrain\Reports**
- Open the SPK file that was sent in the e-mail notification. This action is available only if you set a notification when you defined the model retrain task.

For an example of a comparison report in HTML format, see Appendix 6, "Model Retrain Comparison Report Example," in *SAS Model Manager: User's Guide*.

*Chapter 11*

# Tutorial 10: Using Workflow Console



## Overview of Workflow Console

The SAS Model Manager Workflow Console is an interface to SAS Workflow that you can use to track the progress of a modeling project or version. A SAS Model Manager administrator or a SAS administrator uses SAS Workflow Studio to define process definitions and to make them available to SAS Model Manager for use. Process definitions contain the set of activities, participants, policies, statuses, and data objects that comprise a business task. After the process definitions are made available, a SAS Model Manager administrator uses Workflow Console to create workflow instances to be used with SAS Model Manager. A *workflow instance* is a working version of a workflow process definition. Each workflow instance consists of activities. Instances and activities can contain user-defined properties and comments so that you can share information with other users, or make notes. The status that you select when completing an activity determines the next activity in the workflow process.

From the SAS Model Manager client application, you can view workflow instances, create a new workflow instance for a project or version, and view your workflow inbox to work with activities, depending on the user permissions. The option that is selected and the user permissions determine the category view and content that are displayed when Workflow Console is launched. SAS Model Manager administrators can view the Process Definitions, Instances, and Activities category views of Workflow Console. SAS Model Manager users and advanced users can view only the Activities category view. For more information about user permissions, see “Configuring Users, Groups, and Roles” in Chapter 3 of *SAS Model Manager: Administrator's Guide*.

### *See Also*

Chapter 7, “Using Workflow Console,” in *SAS Model Manager: User's Guide*

# Managing the Workflow Process

## Overview

SAS Model Manager Workflow Console can be used to manage instances of workflow process definitions. A SAS Model Manager administrator can create new workflow instances, view workflow process definitions, and interact with activities that are associated with a workflow instance. If the SAS Model Manager administrator is assigned to the workflow role of business administrator, the administrator can influence the progress of an activity by actions such as assigning an activity, or releasing the activity that is claimed by another user.

In this exercise, you create workflow instances, work with participants, customize category views and terminate workflow instances.

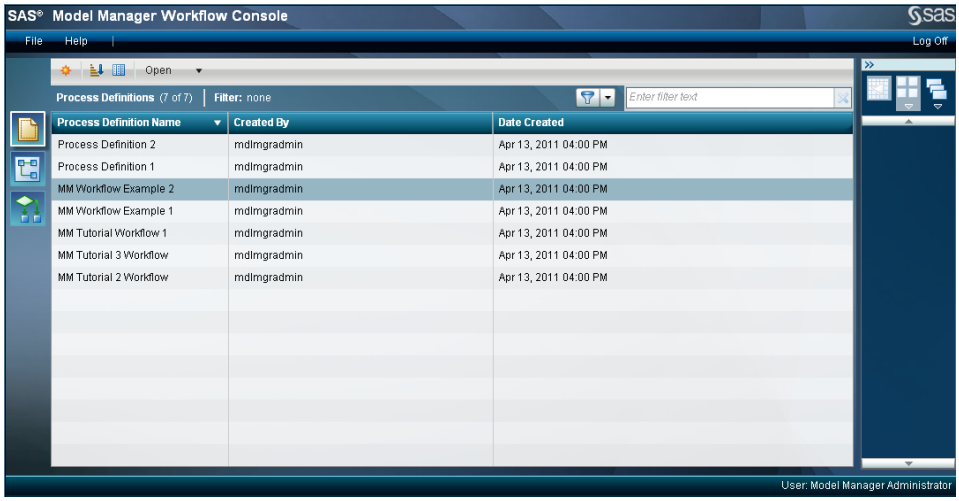
## Prerequisites

The exercises in this tutorial require that you have made the workflow process definitions available to SAS Model Manager. For more information, see "Prepare for Using SAS Workflow" on page 13.

## Create Workflow Instances

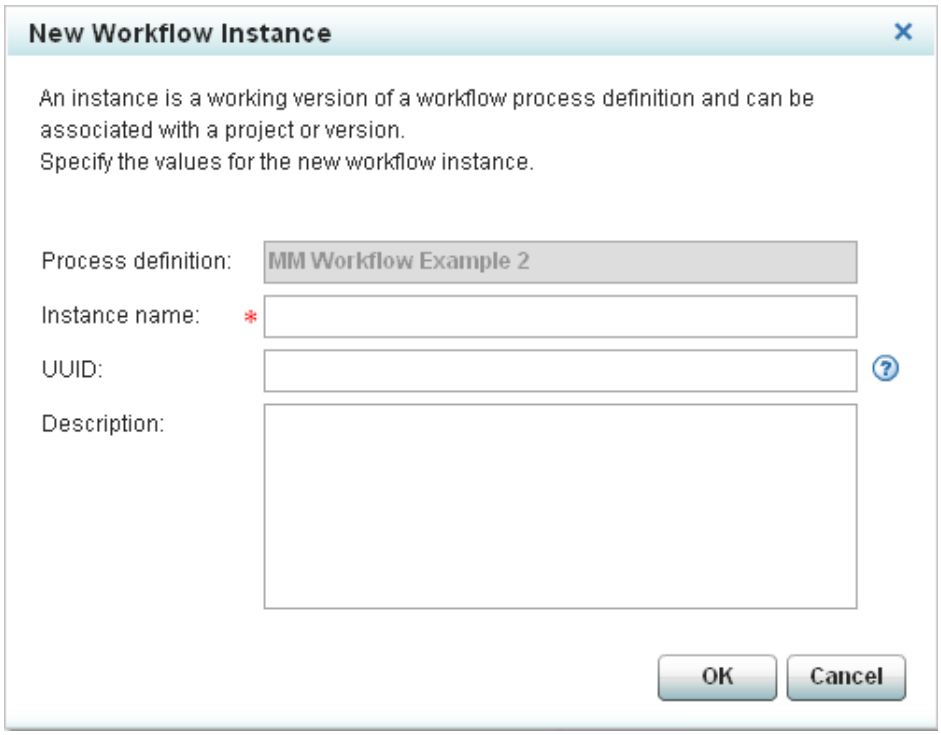
To create a new workflow instance, follow these steps:

1. Log on to SAS Model Manager as a member of the **Model Manager Administrator Users** group.
2. Select **Tools** ⇨ **Manage Workflow**. Workflow Console is launched in a Web browser and displays the Process Definitions category view.



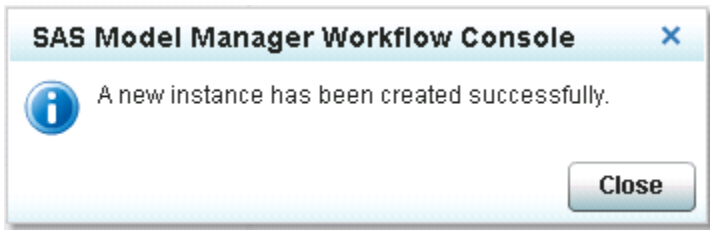

3. Select a process definition (for example, **`MM Workflow Example 2`**) and click . The New Workflow Instance window appears.

   *Note:* The workflow process definitions that have been provided for the tutorials already have participants assigned.

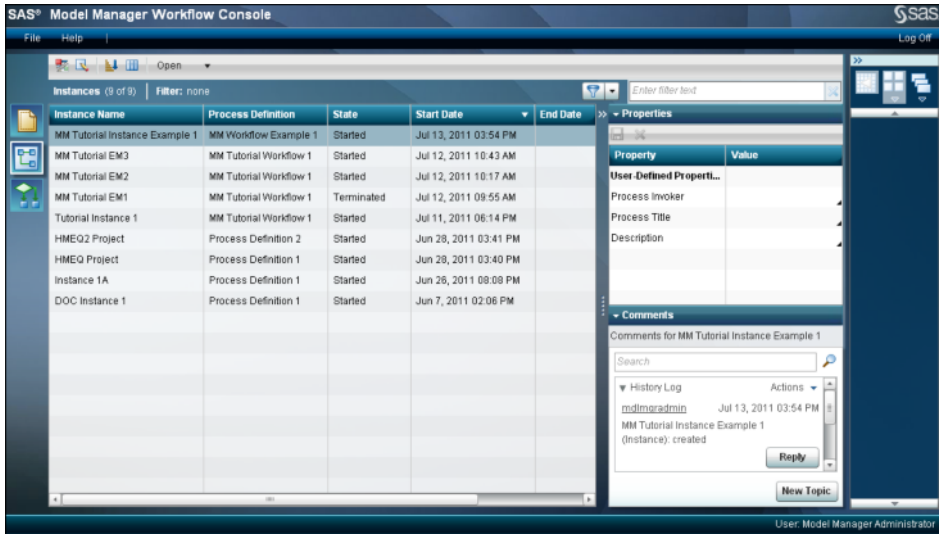



4. Enter a name for the instance (for example, **`MM Tutorial 10 Demo 1`**).
5. For this tutorial, leave the UUID field blank. You can also copy the UUID system property value for a project or version from the Properties view in the SAS Model Manager main window.

   *Note:* The field label and other characters that precede the UUID value must be removed.
6. Enter a description for the instance (for example, **`Instance for tutorial 10`**).
7. Click **OK**. A message appears, indicating that the instance has been successfully created.




8. Click **Close**. The new workflow instance is now available in the Instances category view.
9. Repeat steps 2 through 7 to create at least one additional instance and then continue to the next step.
10. To view the new instances, click  . The Instances category view appears. Select the instance to view information that is associated with the new instance.




## Modify a Workflow Instance

### About Modifying Workflow Instances

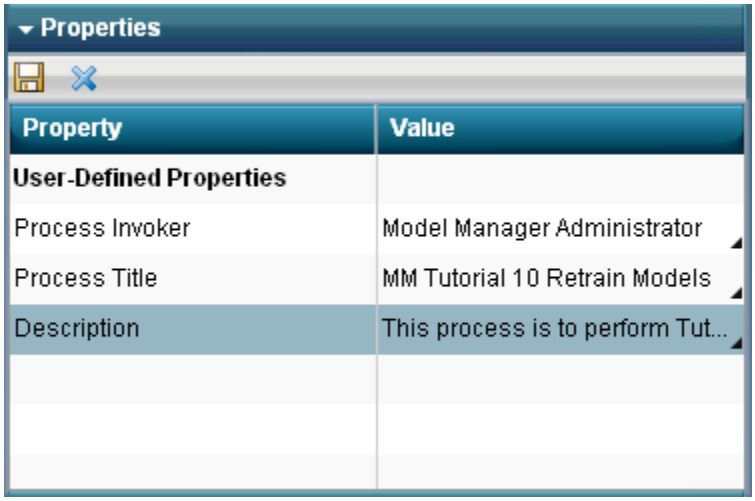
In this exercise, you modify the properties associated with an instance, add comments to an instance and assign participants to activities.

### Modify Instance Properties

To modify the properties associated with an instance, follow these steps:

1. From the Instances category view, select an instance that you created in the previous exercise.
2. In the **Properties** pane, click in the cell for each user-defined property and specify a value.

Here is an example:




3. Click  to save the properties.

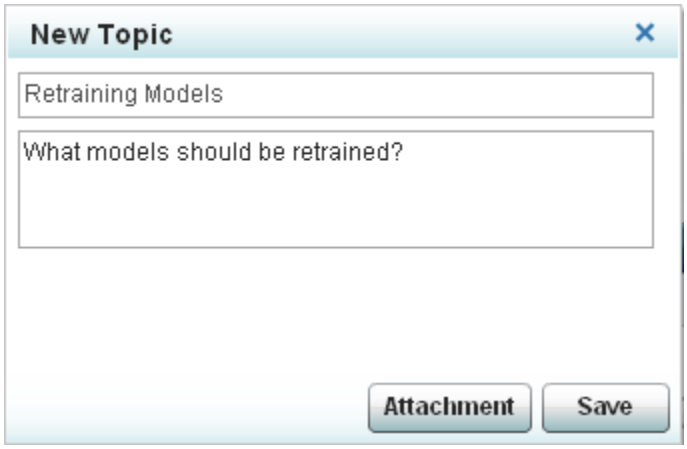
### *Add Comments*

To add comments to the selected instance, follow these steps:

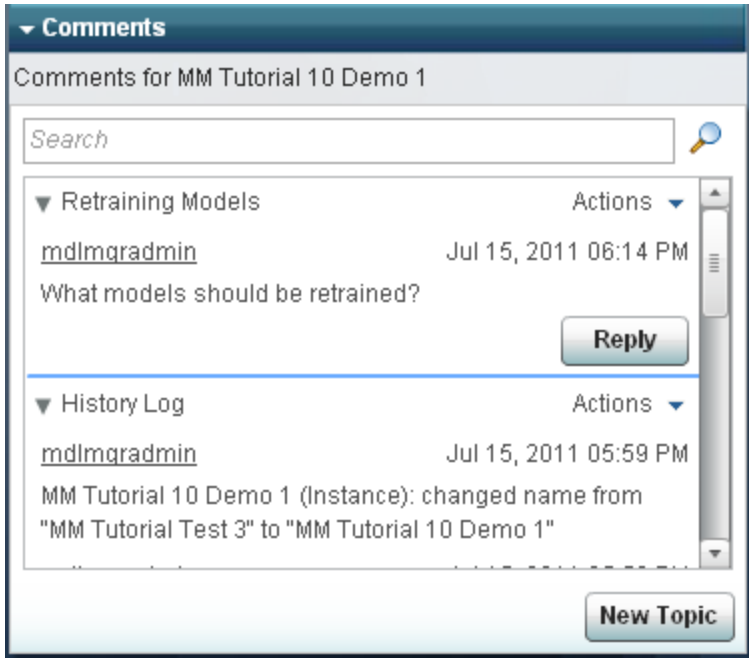
1. In the **Comments** pane, click **New Topic**. The New Topic window appears.


2. Enter a title and comment for the new topic.

   *Note:* For information about adding attachments, see "Working with Comments" in Chapter 7 of *SAS Model Manager: User's Guide*.



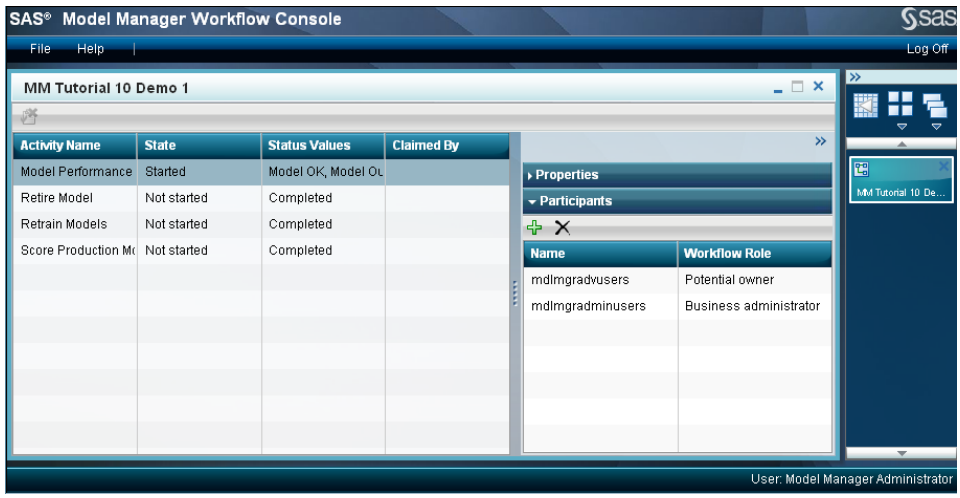

3. Click **Save**. The new topic appears in the comments pane.



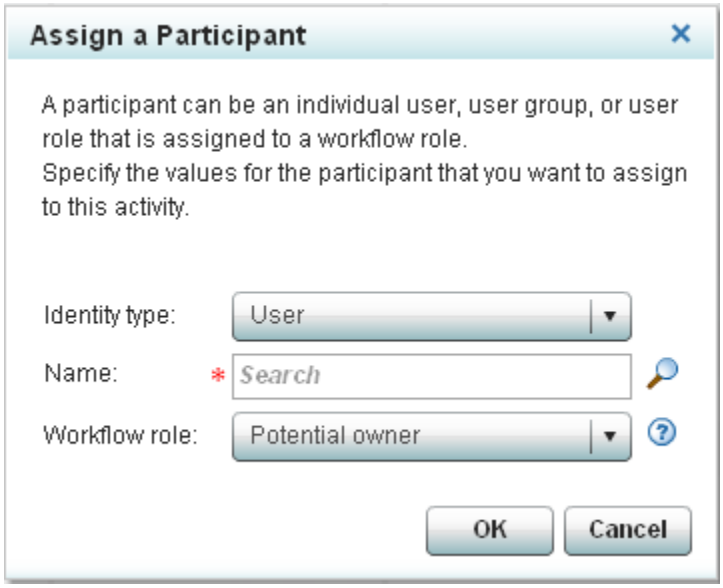

### *Assign Participants*

To assign an additional participant to an activity, follow these steps:

1. Double-click an instance to view the Instance details view, which contains a list of activities that are associated with the workflow instance.



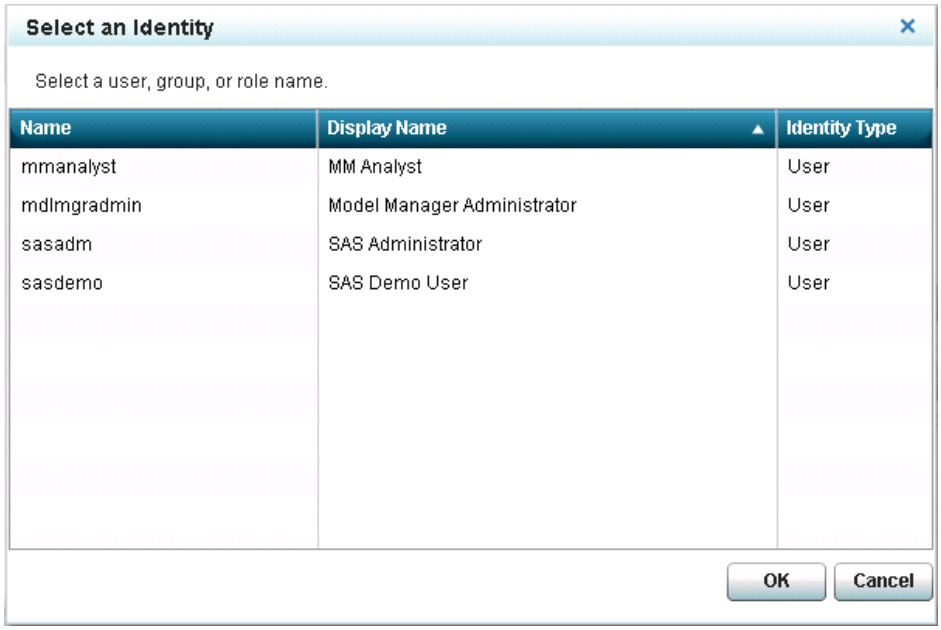

2. Select the **Retrain Models** activity and click + in the **Participants** pane. The Assign a Participant window appears.




3. Select one of the identity types: user, group, or role. The SAS Model Manager user groups that were created by default during installation and configuration, were assigned to each activity when the sample workflow process definition was created.

4. Enter part of the user, group, or role name. For example, you can enter the user name **sasdemo** or part of a user name that exists on the SAS Metadata Server. Then click  .

   *Note:* If you do not enter part of the name, all of the names for the selected identity type are displayed. In addition, if you manually enter a name value and do not click the search button, the name is not verified against the SAS identity participant list.



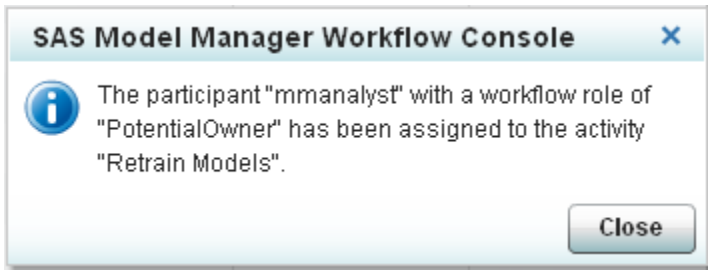

   Select a name and click **OK**.

5. Select a workflow role for the participant.

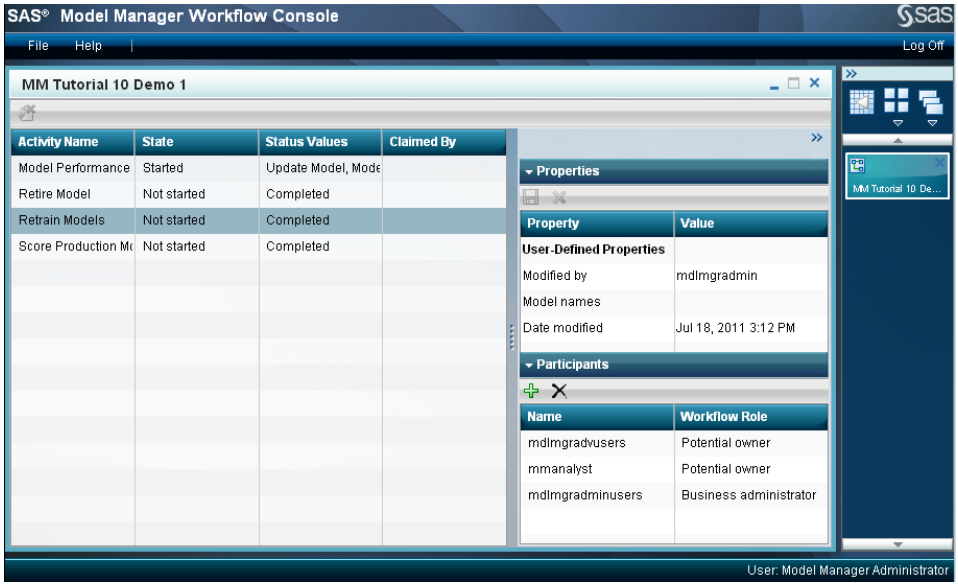
   Here are the workflow roles that you can assign to participants for a workflow activity:

- **Business administrator**: a participant who can influence the progress of an activity by actions such as adding comments, assigning an activity, or releasing the activity claimed by another user.
- **Potential owner**: a participant who can claim an activity in a workflow process and who becomes the actual owner of an activity.

Click **OK**. A message appears, indicating whether the participant was successfully assigned to the activity. For example, the user **`mmanalyst`** was assigned the workflow role of **potential owner**.




6. Click **Close**. Here is an example of the properties and participants associated with the **Retrain Models** activity.




## *Terminate Workflow Instances*

When you terminate a workflow instance, all activities that have not yet been completed in the workflow process are changed to a state of **`Terminated`**. After you terminate a workflow process for an instance, the process cannot be restarted.

To terminate a workflow process, follow these steps:

1. From the Instances category view, select one of the additional instances that you created in the exercise "Create Workflow Instances" on page 166 , and click  .

   *Note:* In order to continue with this tutorial, you need to keep at least one workflow instance active.

2. Click **Yes** to terminate the workflow process for the selected instance.

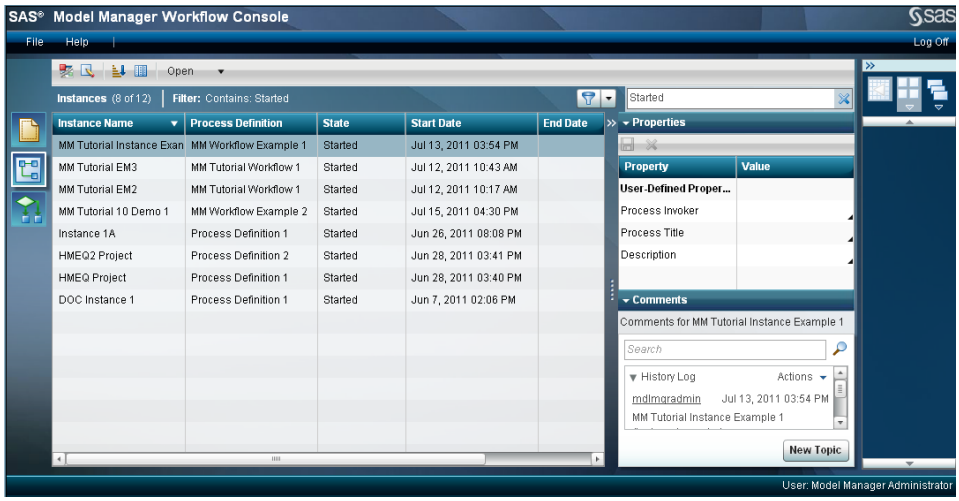
3. Click **Close** to return to the Instances category view.

### *Filter List Content*

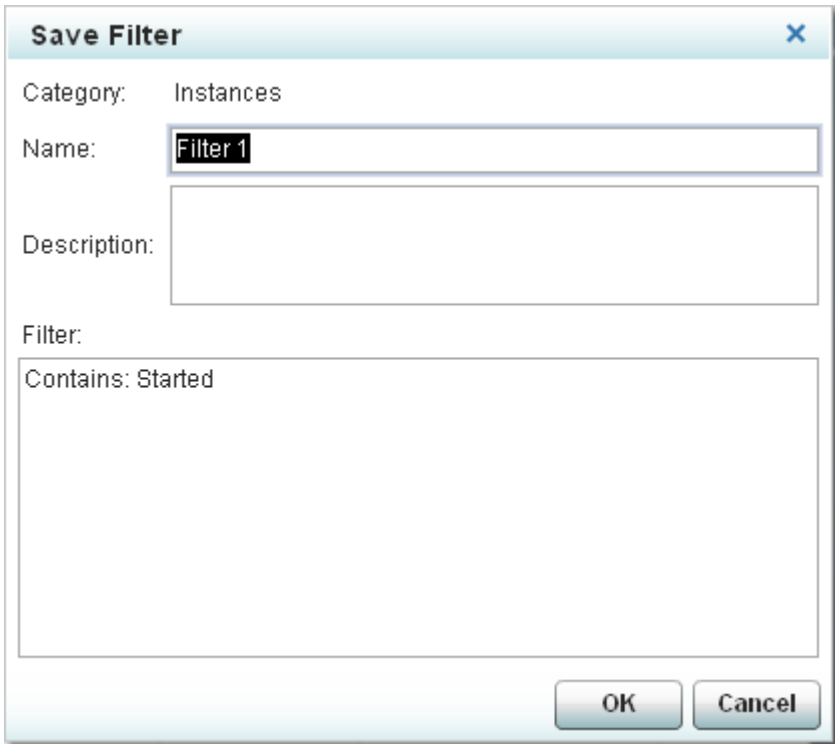
You can filter the list in a category view to display only particular process definitions, instances, or activities. For example, suppose you want to see only instances in the Instances category view that are in the state of **`Started`**.

To filter the content that appears in the list, follow these steps:

1. On the filter bar, specify the filter criteria of **`Started`** in the **Enter filter text** area. Here is an example of the list with the filter applied.




2. To save the filter, click . The Save Filter window appears.



3. In the Save Filter window, specify a name for the filter and an optional description. Click **OK**.

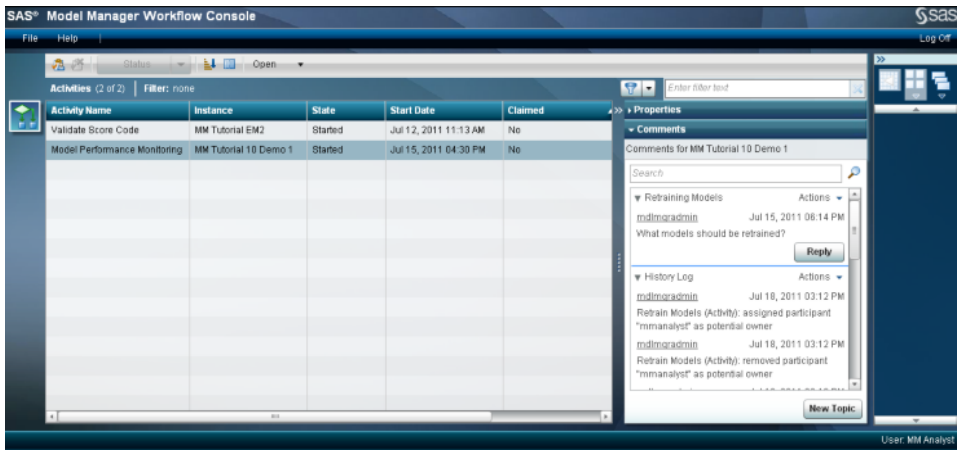
   *Note:* The filter name can be modified in the Manage Filters window. The rule and description cannot be modified for an existing filter. If you want to change the rule or description, you must create a filter that has the same name to replace the existing filter. For more information, see "Filtering List Content" in Chapter 7 of *SAS Model Manager: User's Guide*.

## Working with Workflow Activities

The Activities category view of Workflow Console displays the activities that you have been assigned as potential owner or business administrator, and that have a state of **`Started`**. In this exercise, you claim activities, specify properties, add comments and complete activities.

To complete an activity, follow these steps:

1. Log on to SAS Model Manager as a member of the **Model Manager Advanced Users** group or **Model Manager Administrator Users** group.
2. Select **Tools** ⇨ **My Workflow Inbox** or click . Workflow Console is launched in a Web browser and displays the Activities category view.




3. Select an activity and click to claim an activity.

   *Note:* You can select an activity name and click to release an activity that you had previously claimed. Only a SAS administrator or SAS Model Manager administrator can release an activity that has been claimed by another participant. For more information, see "Releasing an Activity" in Chapter 7 of *SAS Model Manager: User's Guide*.

4. Click on a property value in the Properties pane, and then enter a value or change the existing value.

   *Note:* Not all activities have user-defined properties.

5. To save the changes to the properties, click .

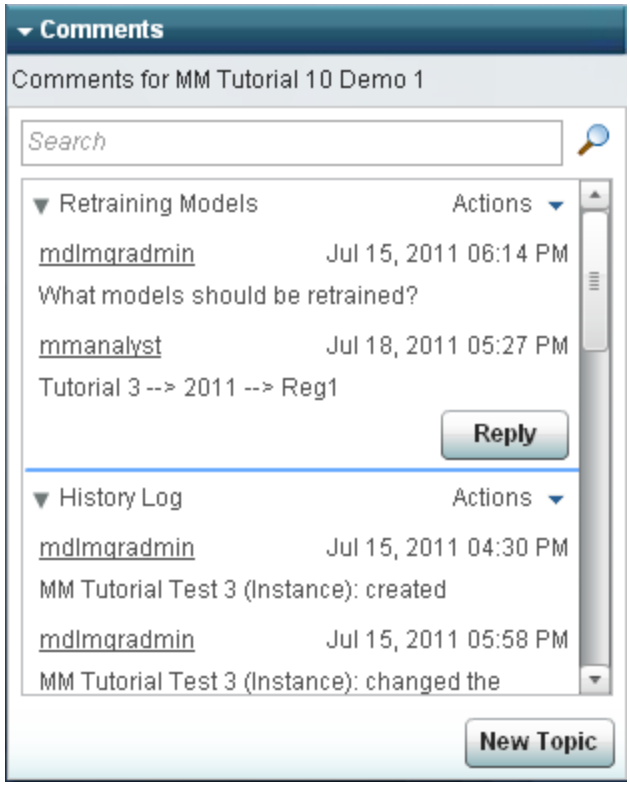
   If you do not want to save the changes to the properties, click .

6. (Optional) Click **New Topic** to add a comment or click **Reply** to add to an existing comment using the **Comments** pane.

Example of the New Topic window:

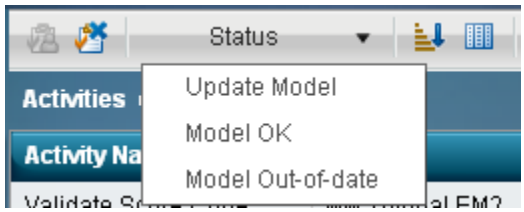

Example of the Reply window:


Click **Save**. The comments now appear in the **Comments** pane.




For information, see “Working with Comments” in Chapter 7 of *SAS Model Manager: User's Guide*.

7. Select a status value to complete the activity. The workflow process continues to the next activity. Here is an example of the **`Model Performance Monitoring`** activity that has three statuses to choose from.




8. Repeat steps 2 through 6 until the workflow process has been completed.

For more information, see “Working with Workflow Activities” in Chapter 7 of *SAS Model Manager: User's Guide*.