SAS® Decision Manager 3.1: Administrator's Guide

Contents

Part 1

Installation and Configuration

Chapter 1
Performing Pre-installation Tasks

Pre-installation Steps

Before you install SAS Decision Manager, be sure to review the Pre-installation Checklist that is provided with your deployment plan. This checklist provides a detailed list of the pre-installation requirements. It also enables you to record important information that you need when you are installing the software.

Perform the following steps before you install SAS Decision Manager:

1. Verify that your system meets the minimum requirements. See the system requirements documentation on support.sas.com.

2. Determine the database that you want to use.

3. Determine the time zone requirements. You might need to synchronize the time zones that are specified in all of your operating environments.

4. Complete the pre-installation steps for your database. See "Pre-installation Tasks for SAS Decision Manager Common Data Server" on page 4 or "Pre-installation Tasks for an Oracle Database" on page 5.

Determine the Database to Use

You can use either Oracle or the SAS Decision Manager Common Data Server for the SAS Decision Manager database.

The SAS Decision Manager Common Data Server is based on PostgreSQL 9.1.9. For more information, see "SAS Web Infrastructure Platform Data Server" in *SAS Intelligence Platform: Middle-Tier Administration Guide* at http://support.sas.com/documentation/intellplatform/index.html.

For Oracle, complete the tasks described in "Pre-installation Tasks for an Oracle Database" on page 5. For SAS Decision Manager Common Data Server, ensure that you have the information listed in "Pre-installation Tasks for SAS Decision Manager Common Data Server" on page 4.

Determine Time Zone Requirements

All of your operating environments (on all tiers in a multi-tier environment) must be set to the same time zone only if your site meets both of these conditions:

• You will deploy rule flows by using SAS Real-Time Decision Manager.

• Those rule flows use terms of type Date or Datetime.

SAS Real-Time Decision Manager uses a custom data type that accounts for differences between time zones when it performs calculations. If the time zones do not match across all of your environments, you should not use Date or Datetime data types in rule flows.

Pre-installation Tasks for SAS Decision Manager Common Data Server

During the installation and configuration of SAS Decision Manager, the SAS Deployment Wizard requires information about the database that SAS Decision Manager uses.

If you are using the SAS Decision Manager Common Data Server that is based on Postgres, you need the information in the following table in order to complete the steps in the SAS Deployment Wizard.

Table 1.1 SAS Deployment Wizard Information for SAS Decision Manager Common Data Server

Property	Description
Database Type	Specifies the database type to use for the SAS Decision Manager database. Select **SAS Decision Manager Common Data Server**.
Database Name	Specifies the database name. The default name for the database is `dcmdb`.

Property	Description
Database User	Specifies the user name for the database administrator. This user owns the database and has superuser privileges. The default user name is **dcmdbowner**.
Database Password	Specifies a password for the user ID that is associated with the database account.
Port	Specifies the port that is used by the database. The default port for SAS Decision Manager Common Data Server is 10482.
Host Name	Specifies the fully qualified host name of the server on which the database is installed.
User ID	Specifies the user name for the user whose credentials will be used to access the SAS Decision Manager Common Data Server database. The default user name is **dcmdb**.

For database-specific information about configuring a database, see *SAS Intelligence Platform: Installation and Configuration Guide*.

Pre-installation Tasks for an Oracle Database

About the Oracle Pre-installation Tasks

If you are using Oracle for your SAS Decision Manager database, perform the following steps before you install SAS Decision Manager:

1. Verify that you have the correct JDBC drivers.

2. Verify that you have a tnsnames.ora file for your Oracle client that corresponds to your database.

3. Determine the required database information.

4. Specify the required database privileges.

5. Test the connection to your database.

Verify JDBC Drivers for Oracle

Verify that you have the correct JDBC drivers. To ensure proper installation of SAS Decision Manager, the drivers must be on each middle-tier server, and they must be in a directory that does not contain any other files.

The JDBC drivers for Oracle are located in the Oracle installation directory. You can also download the ojdbc6.jar file for Oracle Database 11g from http://www.oracle.com/us/downloads/index.html. Select **JDBC drivers**. Download the latest Oracle 11.2*x* driver. The JDBC driver version must match the database version.

For more information about supported database drivers, see the system requirements documents at http://support.sas.com/documentation/installcenter/.

Determine the Information Required for the Oracle Database

During the installation and configuration of SAS Decision Manager, the SAS Deployment Wizard requires information about the Oracle database that SAS Decision Manager uses. Record the information in the following table.

You enter this information in the SAS Decision Manager Database Properties and SAS Decision Manager Database JDBC Properties windows.

Table 1.2   *SAS Deployment Wizard Information for Oracle*

Prompt	Description
Host Name	Specifies the fully qualified host name of the server on which the database is installed.
Port	Specifies the port number that is used by the database. The default port for Oracle is 1521.
Directory containing JDBC driver jars	Specifies the location of the database vendor's JDBC JAR file. This file must be available on the middle tier and on any machine on which you are deploying SAS Decision Manager in order to configure SAS Decision Manager database. See "Verify JDBC Drivers for Oracle" on page 5 for more information.

Database SID or Service Name	Specifies the Oracle database name. The database name must match either the service name or the Oracle site identifier (SID), both of which can be found in the tnsnames.ora file.
	
	If you select **Use Oracle database name as a Service Name**, then you must enter the service name that is specified in the tnsnames.ora file. For example, if you had the following entry in the tnsnames.ora file, you would enter **monitordb** in the **Database SID or Service Name** field:
	
	`monitordb =`
	`  (DESCRIPTION =`
	`    (ADDRESS_LIST =`
	`       (ADDRESS =`
	`          (COMMUNITY = TCP_COMM)`
	`          (PROTOCOL = TCP)`
	`          (HOST = hostname.your.company.com)`
	`          (PORT = 1521)`
	`       )`
	`    )`
	`    (CONNECT_DATA =`
	`       (SERVICE_NAME = monitordb)`
	`    )`
	`  )`
	
	Note: In the tnsnames.ora file, the **Net Service Name** and the **Service Name** fields must be the same.
	
	You can also find the Oracle SID in the tnsnames.ora file. Alternatively, you can run the following query using a database user ID on your Oracle instance:
	
	`select instance from v$thread`

User ID	Specifies the user ID of the database user whose credentials are used to access SAS Decision Manager data on the server.
Password	Specifies the password of the user ID whose credentials are used to access SAS Decision Manager data on the server.

Prompt	Description
Schema Pattern	Specifies the schema name for the database. The default schema is the same as the user ID.

Specify the Required Database Privileges for Oracle

Ensure that the users of your database have the required database privileges. Here are the required privileges for Oracle databases:

- CONNECT
- CREATE SESSION
- RESOURCE
- CREATE TABLE
- CREATE VIEW
- CREATE SEQUENCE
- CREATE TRIGGER

Test the Connection to Your Database

Execute a command from the terminal to verify that your database is set up. For example, to use an Oracle database, you can execute the following command using SQL*Plus:

```
sqlplus USER/PASSWORD@ORACLE_SID
```

You must be able to execute this command from any directory. If you are able to execute a database command such as this only from the database installation directory, then verify that the PATH variable is set up correctly. The database client application must be installed and available on the path specified by the PATH variable.

Chapter 2

Installing SAS Decision Manager

Products Installed with SAS Decision Manager

Your deployment plan for SAS Decision Manager includes additional SAS products that support and complement SAS Decision Manager functionality. See the software order e-mail or the ordersummery.html file that is in your SAS Software Depot at *software_depot*/install_doc/*order_number*/ordersummary.html. The SAS Deployment Wizard prompts you to install and configure each of the products in your deployment plan.

Running the SAS Deployment Wizard

About Running the SAS Deployment Wizard

To run the SAS Deployment Wizard, follow the instructions in "Install and Configure SAS Interactively" in *SAS Intelligence Platform: Installation and Configuration Guide*.

Note: You can run the wizard on operating systems that do not use a windowing environment. For more information, see *SAS Deployment Wizard and SAS Deployment Manager: User's Guide* at http://support.sas.com/documentation/installcenter/94/index.html.

The type and number of configuration-related pages that you see depend on the prompt level that you choose, the SAS tier that you are currently deploying, and the contents of your SAS 9.4 custom order. The following topics provide information for prompts that are specific to SAS Decision Manager. For additional information about any of the SAS Deployment Wizard prompts, see the online Help for the wizard page in question.

Select the SAS Application Server

If you are installing SAS Decision Manager as an add-on product and have already defined other SAS application servers, the SAS Deployment Wizard asks you to select which application server you want to use. Select an application server other than SASMETA.

Configure the Database

During deployment of SAS Decision Manager on SAS 9.4, the SAS Deployment Wizard creates and configures the database tables in the SAS Decision Manager Common Data Server database by default. This database uses the PostgresSQL database management system.

You can use a third-party database server with SAS 9.4. In SAS 9.4, Oracle is the only other third-party database management system that is supported for the SAS Decision Manager database. For information about which versions of the alternative databases are supported, see "Reviewing Third-Party Database Requirements" in *SAS Intelligence Platform: Installation and Configuration Guide*.

The SAS Deployment Wizard prompts you to enter the information that you gathered when you completed the pre-installation tasks for your database. See "Pre-installation Tasks for SAS Decision Manager Common Data Server" on page 4 and "Pre-installation Tasks for an Oracle Database" on page 5 for more information.

For database-specific information about configuring a database, see *SAS Intelligence Platform: Installation and Configuration Guide*.

Create and Load Tables through the SAS Deployment Wizard

If you select **SAS Decision Manager Common Data Server** as the database type, the database tables are automatically created and loaded during the installation and configuration process for the SAS Decision Manager Common Data Server. The default name for the database is `dcmdb`.

If you select **Oracle** as the database type, the **Automatically create tables and load data** check box in SAS Deployment Wizard is enabled. If you want the Oracle database tables to be created and loaded automatically, leave this box selected. If you want to create the tables yourself, then clear the check box, and submit the necessary SQL statements after the wizard finishes running. See "Create Oracle Database Tables" on page 21 for more information.

Chapter 3
Performing Post-installation Tasks

Post-installation Configuration and Verification Steps

After you install SAS Decision Manager using SAS Software Depot, you must perform additional configuration steps before you can use SAS Decision Manager.

1. Verify that all installation and configuration steps in the Instructions.html file have been completed. The Instructions.html file is located in ***sasconfigdir*\\Lev# \\Documents**. Follow the instructions that are provided in this file.

2. Create application users and assign permissions.

3. Verify that the WebDAV folder permissions are set appropriately to allow authorized users to create and edit user-defined templates.

4. (Optional) SAS Decision Manager supports the use of multiple SAS Application Servers to be used as part of a performance definition, a scoring test, or a model retrain definition. To use a SAS Application Server context other than the default **SASApp**, you must add a new workspace server and create a new SAS Application Server context using the SAS Deployment Wizard. For more information, see "Add a New Workspace Server and Create a SAS Application Server Context" on page 18.

5. If the SAS Workspace Server is located in a UNIX environment, you must enable the SAS Workspace Server XCMD option in order to support R model functionality.

6. (Optional) Run the scripts to create Oracle synonyms in the database if you do not want to use the default Oracle schema.

7. If you cleared the **Automatically create tables and load data** check box during installation, then you must manually create and load the Oracle database tables for business rules data and modeling project metadata, including history, job definitions, and job logs. For more information, see "Create Oracle Database Tables" on page 21.

8. In order to publish models from SAS Decision Manager to a database for scoring, additional configuration steps are required to prepare the database. For more information, see "Preparing a Data Management System for Use with SAS Model Manager" in *SAS In-Database Products: Administrator's Guide*.

   Note: If your system is configured for Kerberos authentication, each user must have a valid Kerberos ticket. Also, you must complete post-installation configuration steps to enable users to publish models from the SAS Model Manager application. For more information, see "Configure Users Authenticated by Kerberos" on page 38.

9. Verify that the Certificate Authority certificate is available.

10. (Optional) Configure your deployment to use HTTPS.

11. (Optional) Configure your deployment for single sign-on web authentication.

12. (Optional) Add the Visual Analytics: Data Building and Data Management: Lineage roles to the Decision Manager Users group. See "Administering Group and Role

Membership" on page 63 for more information. These roles enable users to run SAS Visual Data Builder and view lineage information for rule flows and models.

13. (Optional) Configure SAS Workflow.

14. Configure the Decision Builder testing directory.

15. Verify the configuration of the dashboard reports directory on the SAS Workspace Server.

16. Configure the Model Manager Java Services Options.

17. (Optional) Configure the limit for the number of observations for a scoring result set.

18. (Optional) Configure the limit for character strings.

19. Review the Business Rules Manager Web properties in SAS Management Console.

20. (Optional) Modify log file settings.

21. (Optional) If you have a license for SAS Factory Miner and want to integrate with SAS Decision Manager to be able to register models to the model repository, see *SAS Factory Miner: Administration and Configuration*.

For more information about post-installation tasks, see *SAS Intelligence Platform: Installation and Configuration Guide*.

Create Users and Assign Permissions

The SAS Deployment Wizard does not create application users by default. The SAS Administrator must create users in SAS Management Console with the appropriate group and role permissions. Make sure that all users are granted the appropriate permissions to the SAS Workspace Server. For more information, see Chapter 6, "Configuring Users, Groups, and Roles," on page 57.

In a Windows environment, each user or group must be granted permission to the **Log on as a batch job** local security policy. This permission is required in order to access functionality in the Data category. For more information, see "Create Windows Operating System Accounts and Groups for Users" on page 14.

In a UNIX environment, all SAS Decision Manager users must be part of a group that has the appropriate group permissions. For more information, see "Create UNIX Operating System Accounts and Groups for Users" on page 15.

Creating Operating System Accounts for Product Administrators and Users

About the User Accounts for SAS Decision Manager

SAS Decision Manager provides two types of user accounts:

Product administrator
    A SAS Decision Manager administrative user is specific to SAS Decision Manager. A product administrator account is not the same as a general administrator account, such as the SAS Administrator (sasadm@saspw). These users must have a valid host operating system account, and you must associate that account with a metadata user.

Users of SAS Decision Manager

These users must have a valid host operating system account, and you must associate that account with a metadata user through SAS Management Console.

You must create the operating system account for the administrator and for regular user accounts as a post-installation task. For more information, see the following topics:

- "Create Windows Operating System Accounts and Groups for Users"
- "Create UNIX Operating System Accounts and Groups for Users"
- Chapter 6, "Configuring Users, Groups, and Roles,"

Using Operating System Groups to Assign Permissions

Users have different operating system privileges on the SAS Workspace Server. By defining a user group for SAS Decision Manager, you can assign all users to the same group and grant the same permissions to all users at one time. All SAS Decision Manager users must have Read, Write, and Execute permissions for each environment directory that a user is permitted to use. Users also need permissions to all of the files and directories in an environment directory. The operating system must be configured to grant these permissions when new files and directories are created. The steps that you follow to do this depend on which operating system groups are defined and your site's security policies.

Create Windows Operating System Accounts and Groups for Users

On the SAS Workspace Server, create an operating system account for the administrator of SAS Decision Manager and all SAS Decision Manager users.

If the SAS Workspace Server is running in the Windows operating environment, use one of the following methods to create this operating system account:

- If you are using an LDAP server to manage your users, define the user (for example, *domain\username*) on the Active Directory server.
- If you are working on a local machine, complete these steps to create this user account:

  1. Right-click the Computer icon on your desktop and select **Manage**. The Computer Management window appears.

     Note: If you are creating users on a server, you can use the Server Manager.

  2. In the left navigation pane, expand the **Local Users and Groups** node. The **Users** and **Groups** nodes appear.

  3. Right-click the **Users** node and select **New User**. The New User window appears.

  4. In the New User window, complete these tasks:

     - Specify a user name and password.

       Note: In Windows, you cannot enter *<domain>\username* (you enter the user name only), but you must enter *<domain>\username* in the SAS Deployment Wizard and SAS Management Console.

     - Clear the **User must change password at next logon** check box.

     - Select the **User cannot change password** check box.

     - Select the **Password never expires** check box.

Click **Create**.

5. Click **Close** to close the New User dialog box.

6. If you want to add the users that you created to a group, perform the following steps:

   a. Right-click the **Groups** node in the Computer Management window, and select **New Group**.

   b. Click **Add**. Enter the user names, separated by semicolons, and click **Check Names**.

   c. Click **OK**.

7. Assign the security policy of **Log on as batch job** for each user or group.

   a. Select **Start** ⇨ **Control Panel** ⇨ **System and Security** ⇨ **Administrative Tools** ⇨ **Local Security Policy**.

   b. From the Local Security Policy window, expand the **Local Policies** node and select **User Rights Assignment**. Then double-click the **Log on as batch job** policy.

   c. Click **Add user or Group**. Enter the user names or group names, separated by semicolons, and click **Check Names**.

   d. Click **OK**.

Create UNIX Operating System Accounts and Groups for Users

Conditions for the User Group

If you are working in a UNIX operating environment, the following conditions must be met:

- A group of users is created for the UNIX operating environment. The logon IDs for each user must be in this group. The group must also include any user who might run code that is created by SAS Decision Manager in a SAS session.

- Users can be members of multiple groups, but the SAS Decision Manager group is the primary group for each user.

- The SAS scripts are updated to grant permissions to the SAS Decision Manager users on the SAS Workspace Server. For more information, see "Update the SAS Scripts to Grant Permissions to the User Group" on page 15.

- Each environment directory has the correct ownership, and the user group has Read, Write, and Execute permissions.

Update the SAS Scripts to Grant Permissions to the User Group

Using the `umask` option, you can grant permissions to users on a conditional basis if the user is part of the SAS Decision Manager user group.

Note: This example might require changes to fit your server configuration. In particular, this example could result in changed permissions on other SAS files, such as OLAP cubes. For example, if you are working with multiple UNIX groups and have a SAS OLAP Server, you must ensure that the account under which the SAS OLAP Server runs has Read and Execute permissions to OLAP files.

To set these permissions:

1. On each SAS Workspace Server, open **/*sasconfigdir*/Lev1/SASApp/ appservercontext_env_usermods.sh**.

2. Enter the configuration information for your operating environment. Here is the general format of this code:

   Note: The following code uses grave accents and not quotation marks.

   ```
   CMD=<your-operating-system-path>
   CURR_GID=`eval $CMD -g`
   GID=<solution-group-id>
   if [$CURR_GID -eq $GID]; then umask 002 fi
   ```

   1 In the **CMD=<*your-operating-system-path*>**, specify the full path on your server where the ID command is stored. You can get this information by entering a **which id** or **whence id** command on your console.

   2 In the **GID=<*solution-group-id*>**, specify the group ID. Type **id** on your console to get the GID and UID information.

   3 A value of 002 is recommended for the **umask** option.

   Here are code examples for each UNIX environment where SAS Decision Manager is supported:

Operating Environment	Sample Code
AIX	```CMD=/usr/bin/id CURR_GID=`eval $CMD -g` GID=201 if [$CURR_GID -eq $GID]; then umask 002 fi```
H64I (HP-Itanium)	```CMD=/usr/bin/id CURR_GID=`eval $CMD -g` GID=201 if [$CURR_GID -eq $GID] ; then umask 002 fi```
S64 (Solaris)	```CMD=/usr/xpg4/bin/id CURR_GID=`eval $CMD -g` GID=201 if [$CURR_GID -eq $GID] ; then umask 002 fi```
SAX (Solaris for X64)	```CMD=/usr/xpg4/bin/id CURR_GID=`eval $CMD -g` GID=201 if [$CURR_GID -eq $GID] ; then umask 002 fi```
LNX (Linux)	``` #!/bin/bash CMD=/usr/bin/id CURR_GID=`eval $CMD -g` GID=500 if ["$CURR_GID" -eq "$GID"] ; then umask 002 fi```

Verify WebDAV Folder Permissions for User-Defined Templates

During the SAS Decision Manager installation process, the **ModelManager**, **sasfolders**, and **sasdav** WebDAV folders are automatically created with default permissions on the SAS Content Server. If you migrated or upgraded from a previous release, the WebDAV folder permissions should be preserved. Use the SAS Content Server Administration Console (SCS Admin Console) to control access to an existing WebDAV folder.

Here are the default permissions for a new installation of SAS Decision Manager for the folder **/ModelManager/ConfigTemplates/ext**:

Group	Permissions	Description
mdlmgrusers	Read	Model Manager Advanced Users group
mdlmgradminusers	Read, Write, and Delete	Model Manager Administrator Users group
mdlmgradvusers	Read	Model Manager Users group

In order to create new model and report templates, as well as edit existing user-defined templates, users must have Read and Write permissions to the **/ModelManager/ConfigTemplates/ext** folder.

To verify or modify the folder permissions for the **ext** folder:

1. Access the SAS Content Server Administration console by entering the following URL in your web browser and substituting the server name and port number of your SAS Content Server: **http://*server-name:port*/SASContentServer/dircontents.jsp**.
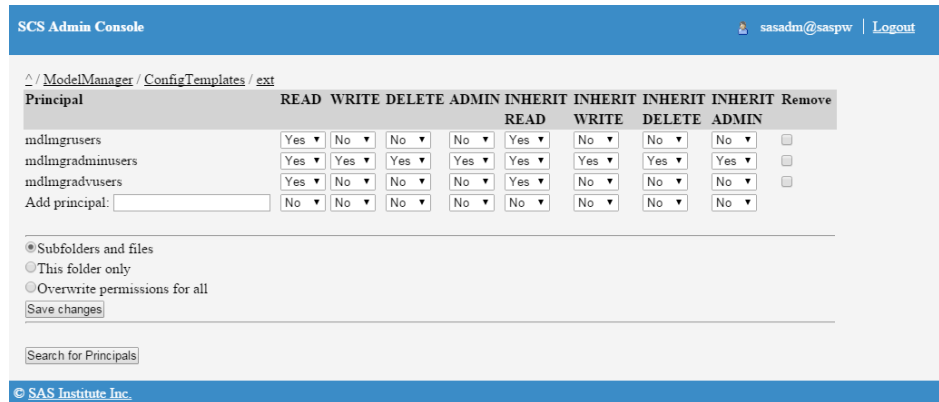
   Note: The default port number for the SAS Web Application Server is 7980 for a UNIX environment and 80 for a Windows environment.

2. Sign in to the console as an unrestricted user (for example, SAS Administrator). The SCS Admin Console window appears.
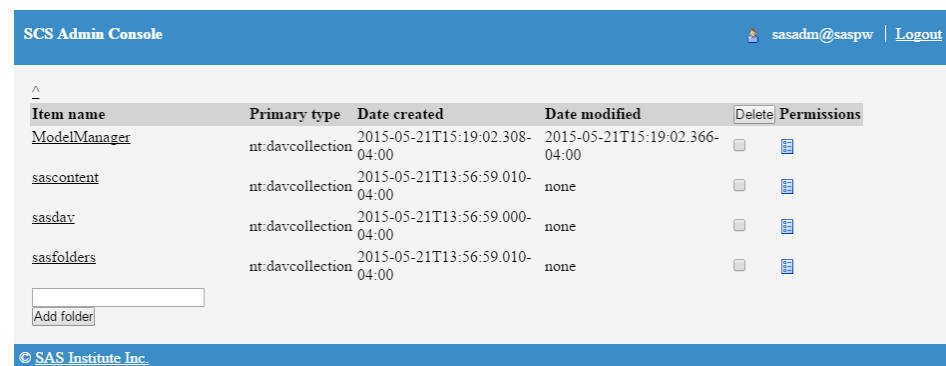
3. Click **ModelManager** ⇨ **ConfigTemplates**.

4. Click the permissions icon ▦ for the `ext` folder. The Permissions page appears.

5. Modify the permissions for the `mdlmgradvusers` principal to give Write permissions by changing the **WRITE** and **INHERIT WRITE** permissions to **Yes**.

   Note: It is recommended to give Delete permissions only to the mdlmgradminusers group.

   For more information, see "Modify Permissions for WebDAV Folders and Files" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Add a New Workspace Server and Create a SAS Application Server Context

SAS Decision Manager on SAS 9.4 provides support for multiple SAS Application Servers that can be used when specifying a performance definition, a scoring test, or a model retrain definition.

To add a new workspace server to your SAS Decision Manager installation:

1. On the machine where the SAS Application Server was installed, start the SAS Deployment Wizard from the highest-level directory in your SAS Software Depot using the command file that is appropriate for your operating system. On Windows systems that command file is setup.exe. For UNIX systems, use setup.sh.

2. Select **Install SAS Software**. Click **Next**.

3. Select **Perform a Planned Deployment**, deselect **Install SAS Software**, and select only **Configure SAS Software**. Click **Next**.

4. You must use the same deployment plan that was used when you installed the original server component. This example uses one of the standard deployment plans. Choose **Select a standard deployment plan**, and select the appropriate **Model Mgr** plan for your machine configuration. Click **Next**.

5. Select the server machine on which you want to configure the SAS Application Server. Click **Next**.

6. On the **Checking System** page, click **Next**.

7. On the Select Configuration Prompting Level page, select **Typical**. Click **Next**.

8. On the Specify Configuration Information page, you must select the same options as your original installation. For this example, use the default settings. Click **Next**.

9. The SAS Deployment Wizard warns you that the configuration directory and level that you specified contain existing files. Select **Yes**.

10. On the Select Products to Configure page, select **SAS Workspace Server**. Click **Next**.

11. On the Local Machine Name page, accept the default settings. Click **Next**.

12. On the Migration Information page, no migration should be performed. Click **Next**.

13. On the Integrated Windows Authentication page, accept the default settings. Click **Next**.

14. On the SAS Metadata Server page, accept the default settings. Verify that the SAS Metadata Server and port are valid. Click **Next**.

15. On the Deployment Accounts: Type of Accounts page, select **Use SAS internal accounts when appropriate**. Click **Next**.

16. On the SAS Internal Account: Unrestricted Administrator page, enter the same password that was used in the original installation. Click **Next**.

17. On the Deployment Accounts: First User page, there is no need to set up a first user ID. Click **Next**.

18. On the Create SAS Application Server Context page, select **Create a SAS Application Server Context**. Click **Next**.

19. On the SAS Application Server: Server Context page, enter `SASApp_2` in the field **SAS Application Server Context Name**. Click **Next**.

20. On the SAS Server Dependencies page, confirm that the value of **SAS Application Server Context** is **SASApp_2**. Click **Next**.

21. On the SAS Workspace Server page, enter the workspace server port. This example uses the default value, 8592. Click **Next**.

22. Select the deployment summary and click **Start**. When the SAS Deployment Wizard is finished, note the additional resources and click **Finish**.

Next you must configure the Job Execution Service and restart the SAS web application servers.

1. Run SAS Management Console and select the **Plug-ins** tab. Select **SAS Application Management ⇨ Configuration Manager ⇨ SAS Application Infrastructure ⇨ Web Infra Platform Services 9.4 ⇨ JobExecutionService**.

2. Right-click **JobExecutionService** and select **Properties**.

3. Select the **Settings** tab in the JobExecutionService Properties window and locate the **Configure Execution Queues from Available Server Contexts** section.

4. Select the new server (for example, **SASApp_2**). Then move it to the **Selected** list.

5. In the **SASApp_2 Execution Queue Properties** section, clear **Enable for interactive execution?**.

6. Click **OK**.

7. Restart SAS servers, including the SAS Metadata Server and SAS Web Application Servers. For example, in a default Windows installation this can be done for Services from the Windows Management Console by restarting the Service name, such as **SAS [Config-Lev1] WebAppServer SASServer11_1**.

To verify your new server from SAS Decision Manager:

1. Sign in to SAS Decision Manager.

2. Create a scoring test that uses the new SAS Application Server context.

3. Execute the scoring test.

For more information, see:

- "Managing SAS Application Servers" in *SAS Intelligence Platform: Application Server Administration Guide*

- "Managing Workspace Servers and Stored Process Servers" in *SAS Intelligence Platform: Application Server Administration Guide*

- "Add a New Logical Server in an Existing SAS Application Server" in *SAS Intelligence Platform: Application Server Administration Guide*

- "Job Execution Service" in *SAS Intelligence Platform: Middle-Tier Administration Guide*

Enable the SAS Workspace Server XCMD Option

When you are running the SAS Workspace Server in a UNIX environment for SAS Decision Manager 3.1 on a SAS 9.4 deployment, the XCMD option is turned off by default. Therefore, you cannot use the SYSTEM function, the X command, or the PIPE option in a FILENAME statement. You must enable the SAS Workspace Server XCMD option in order to support R model functionality.

To enable the XCMD option:

1. From SAS Management Console, expand the **Environment Management ⇨ Server Manager** node on the **Plug-ins** tab.

2. Select and expand **SASApp ⇨ SASApp – Logical Workspace Server**.

3. Right-click **SASApp - Workspace Server** and select **Properties**.

4. Select **Options ⇨ Advanced Options ⇨ Launch Properties** and then select the **Allow XCMD** check box.

5. Click **OK** to save the setting.

6. (Optional) If you have multiple SAS Application Servers with the server type of Workspace Server, repeat steps 2 through 5.

7. Restart your SAS Object Spawner.

Create Oracle Database Synonyms

If you use Oracle for your SAS Decision Manager database and you do not want to use the default schema, you can run two SQL scripts to create synonyms for the database tables. These scripts are in
SASHome\SASDecisionManagerCommonMidTierforDecisionManager\3.1\ Config\Deployment\dbscript\Oracle\optional. Use your preferred Oracle tool to run these scripts. One of these scripts, brm_grant_priv_synonym.sql, uses

substitution variables. If your Oracle tool does not support substitution variables, then you need to manually replace the variable with its value, as described in Step 2.

To run these scripts:

1. In the script named brm_grant_priv_synonym.sql, find the following line:

    ```
    def usernm='YOUR_USER_NAME';
    ```

    Replace **YOUR_USER_NAME** with the user ID that you are using to access the SAS Decision Manager database.

    If your Oracle tool supports substitution variables, skip to Step 3. If not, continue with Step 2.

2. If your Oracle tool does not support substitution variables, in the script named brm_grant_priv_synonym.sql, replace all occurrences of **&usernm** with the user ID that you are using to access the database.

3. Run the script named brm_create_synonym.sql using your preferred Oracle tool. This script does not require substitution variables.

4. Run brm_grant_priv_synonym.sql using your preferred Oracle tool.

Create Oracle Database Tables

When you ran the SAS Deployment Wizard, the **Automatically create tables and load data** check box was selected by default for the SAS Decision Manager database. (See "Create and Load Tables through the SAS Deployment Wizard" on page 10.) If you cleared the **Automatically create tables and load data** check box and you do not have an existing database instance, you must run the SQL scripts to create and load the tables.

Run the CreateMMTables.sql script in order to create and load the model tables. The script is located in **SASHome\SASModelManagerMidTier\14.1\Config \Deployment\Content\dbscript\oracle**.

Run the following scripts to create the business rules tables with a compatible database client for your installation. These scripts are located in **SASHome\SASDecisionManagerCommonMidTierforDecisionManager\ 3.1\Config\Deployment\dbscript\oracle**.

Before you run these scripts, replace **@schema.name@** in each file with the schema name for your database.

1. brm_create_table.sql

2. brm_create_sequence.sql

3. brm_create_constraint.sql

4. brm_create_view.sql

5. brm_required_inserts.sql

6. edm_workflow_interface_create_table.sql

7. edm_workflow_interface_create_sequence.sql

8. edm_workflow_interface_create_constraint.sql

9. edm_workflow_interface_create_trigger.sql

10. edm_workflow_interface_required_inserts.sql

11. edm_create_table.sql

12. edm_create_constraint.sql

13. edm_create_sequence.sql

14. edm_required_inserts.sql

15. dcb_create_table.sql

16. dcb_create_constraint.sql

17. dcb_required_inserts.sql

Verify the Certificate

During installation and configuration of SAS 9.4, the SAS Deployment Wizard enables you to configure the SAS Web Server to use HTTPS and Secure Sockets Layer (SSL) certificates automatically. Verify that the Certificate Authority certificate is available to the trust store for the browser and Java clients such as SAS Workflow Studio and SAS Management Console. For more information, see the Instructions.html file in the directory ***SASConfigDir*\\Lev1\\Documents** and *SAS Intelligence Platform: Middle-Tier Administration Guide* for information about validating your environment.

If you did not use the SAS Deployment Wizard to configure the SAS Web Server to use HTTPS and SSL certificates, you can configure it manually. For more information, see "Configuring SAS Web Server Manually for HTTPS" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

The communication path between SAS Web Server and SAS Web Application Server uses HTTP by default. If you configured the SAS Web Server to use HTTPS by using the SAS Deployment Wizard, you must complete additional steps in order to use HTTPS between SAS Web Server and SAS Web Application Server. For more information, see "Configuring SAS Web Application Server to Use HTTPS" in *SAS Intelligence Platform: Middle-Tier Administration Guide*. If you configure the SAS Web Application Server to use HTTPS, see "Configure Your Deployment for HTTPS" on page 22 for additional instructions.

Configure Your Deployment for HTTPS

The steps listed in "Configuring SAS Web Application Server to Use HTTPS" in *SAS Intelligence Platform: Middle-Tier Administration Guide* change the communication path between the SAS Web Server and SAS Web Application Server to use HTTPS. If you completed those steps and configured the communication path to use HTTPS, then you must also configure the SAS Web Application Server to use HTTPS.

To configure the SAS Web Application Server to use HTTPS:

1. Edit the sasenv file for your operating environment.

Environment	File	Directory
UNIX	sasenv.sh	**/*config-dir*/Levn/Web/WebAppServer/ SASServer7_1/bin**

Environment	File	Directory
Windows	sasenv.bat	***config-dir*\Levn\Web\WebAppServer \SASServer7_1\bin**

Add the following options to the JVM_OPTS line:

```
-Dbrm.midtier.use.https=true
-Ddcm.midtier.use.https=true
```

2. In Windows environments, edit the wrapper.conf file in the ***config-dir*\Levn \Web\WebAppServer\SASServer7_1\conf** directory.

   Add the following line to the Java Additional Parameters section:

   ```
   wrapper.java.additional.n=--Ddcm.midtier.use.https=true
   ```

   The number *n* is the next number in the sequence of wrapper parameters.

3. Restart SASServer7.

   Note: The options are needed only on SASServer7; they are not needed on SASServer11.

Configure Your Deployment for Single Sign-On Web Authentication

SAS 9.4 web applications can be configured to use single sign-on web authentication. For more information, see the steps in "Web Authentication" in *SAS Intelligence Platform: Middle-Tier Administration Guide*. After you complete those steps, an additional configuration is needed. You must apply the latest SAS Model Manager 14.1 hot fix (V85008 or later) for the new single sign-on configuration to be available. For a list of the latest hot fixes, see SAS Model Manager 14.1 Hot Fixes.

To configure SAS Decision Manager to use single sign-on web authentication:

1. Edit the setenv file for your operating environment.

Environment	File	Directory
UNIX	setenv.sh	**/*config-dir*/Levn/Web/WebAppServer/ SASServer11_1/bin**
Windows	setenv.bat	***config-dir*\Levn\Web\WebAppServer \SASServer11_1\bin**

Add the following option to the JVM_OPTS line:

```
-DMMSingleSignOn=on
```

2. In Windows environments, edit the wrapper.conf file in the ***config-dir*\Levn \Web\WebAppServer\SASServer11_1\conf** directory.

   Add the following line in the Java Additional Parameters section:

```
wrapper.java.additional.n=-DMMSingleSignOn=on
```

The number *n* is the next number in the sequence of the wrapper parameters.

3. Restart SASServer11.

   Note: The option is needed only on SASServer11; it is not needed on SASServer7.

Configure the Decision Builder Testing Directory

The default directory for decision testing results is ***sasconfigdir*\Lev#\AppData \DecisionBuilder\tests**. This directory path is specified by the Decision Manager 3.1 property **dcb.testlib.dir**.

Note: For deployments with multiple application servers configured for Job Execution Service, it is possible for decision test results (log and output) to be saved separately from the code that is generated by a test. If the test runs successfully but the testing results are not available through the Decision Builder user interface, the results might be saved on a different server. To avoid this issue, change the **dcb.testlib.dir** property on each server to specify the testing results directory with a UNC path or a network location. The **dcb.testlib.dir** property on each server must specify the same directory.

To configure the default testing results directory:

1. (Optional) Change the directory location. Complete the following steps in SAS Management Console:

   a. On the **Plug-ins** tab, select **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

   b. Right-click **Decision Manager 3.1** and select **Properties**. The **Decision Manager 3.1 Properties** dialog box appears.

   c. Click the **Advanced** tab, and change the value of the **dcb.testlib.dir** property.

      Note: The testing results directory is automatically created when a user with the correct permissions runs the first decision test.

2. Verify that users have Read, Write, and Execute permissions to the testing directory.

   Note: In a UNIX environment, all users must be part of a group that has the appropriate group permissions. For more information, see "Create UNIX Operating System Accounts and Groups for Users" on page 15 and Chapter 6, "Configuring Users, Groups, and Roles," on page 57.

Configure the Dashboard Reports Directory

In SAS Decision Manager, the dashboard reports directory is configured during installation. The default directory is ***SASConfigDir*\Lev#\AppData \SASModelManager14.1\Dashboard**.

Note: When the SAS Web Application Server and the SAS Workspace Server are located on different physical machines, the Software Deployment Wizard creates a directory on the SAS Web Application Server machine and uses that value for the **App.DashboardDir** property value. You must create a directory that is accessible by the SAS Workspace server, and the SAS Decision Manager users must have permissions to that directory.

To configure a different directory to store the SAS Decision Manager dashboard reports:

1. Connect to the SAS Workspace Server.

2. Create a new directory (for example, `C:\Dashboard`).

   Note: The directory must be located on the SAS Workspace Server or on 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 add (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 add (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 Decision Manager users must be part of a group that has the appropriate group permissions. For more information, see "Create UNIX Operating System Accounts and Groups for Users" on page 15 and Chapter 6, "Configuring Users, Groups, and Roles," on page 57.

4. From SAS Management Console, expand the **Application Management** node on the **Plug-ins** tab.

5. Select and expand **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

6. Right-click **Model Manager JavaSvcs 14.1** and select **Properties**.

7. (optional) Click the **Settings** tab and then select **Report Options**. Use this setting to specify the styles that are available when a user generates dashboard reports, and to enable the indicator override option for defining dashboard report indicators. When you use the indicator override configuration, indicators with conditions are available when you add dashboard report indicators using SAS Decision Manager. For more information, see "Report Options" on page 27.

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**.

10. For changes to take effect, you must restart the web application server.

Configuring Model Manager Java Services Options

Overview of Configuring Model Manager Java Services Options

The **Model Manager Java Services Options** setting in SAS Management Console enables you to modify model management configurations. The configurations can be modified for reporting, for metadata tables usage when publishing a scoring function, for SAS code debugging, SAS system options, In-database connection and publishing options, and performance options for the SAS High-Performance Analytics procedures.

To modify the settings for the report options:

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** ⇨ **Enterprise Decision Manager 3.1**.

3. Right-click **Model Manager JavaSvcs 14.1** and select **Properties**.

4. Click the **Settings** tab and then select **Model Manager Java Services**.
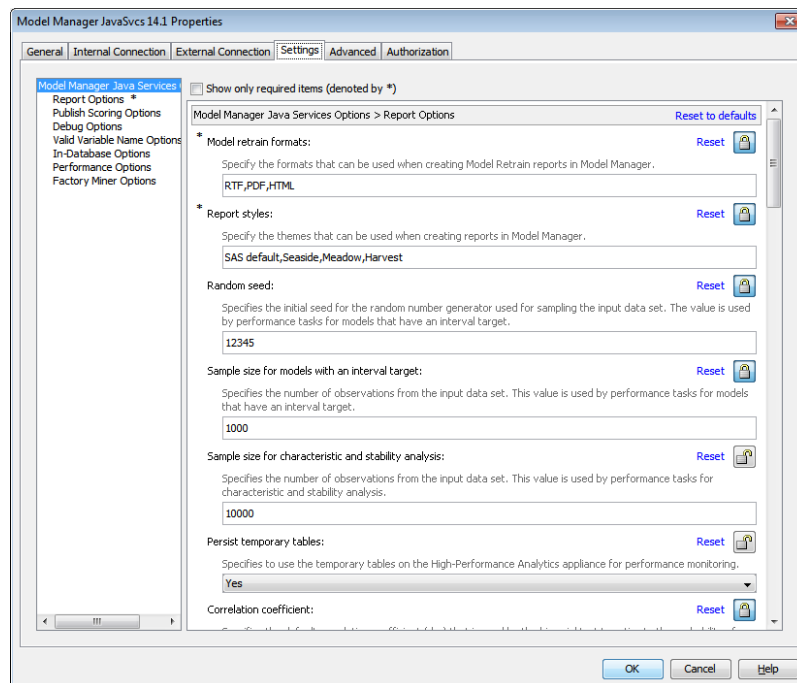
5. Select one of the following options to view and configure the available settings.

   - **Report Options**

   - **Publish Scoring Options**

   - **Debug Options**

   - **Valid Variable Name Options**

   - **In-Database Options**

- • **Performance Options**
- • **Factory Miner Options**

6. Click **OK**.

7. For changes to take effect, you must restart the web application server.

Report Options

The **Report Options** setting in SAS Management Console enables you to modify the SAS Decision Manager configurations for the dashboard reports, for model retrain reports, and for performance monitoring. These reports are available on the **Reports** page for a project in SAS Decision Manager.

To modify the report options setting:

1. Specify the formats that are available when a user creates model retrain reports. The default values are RTF, PDF, and HTML. You can remove any of the default values so that they are not available in SAS Decision Manager.

2. Specify the report styles that are available when a user creates the model retrain reports and dashboard reports. You can add or remove SAS styles. The default values are SAS default, Seaside, Meadow, and Harvest. For more information about SAS styles, see "Style Templates" in *SAS Output Delivery System: User's Guide*.

3. Specify a value for a random seed to be used by performance definitions for models that have an interval target. The default value is **12345**. The **random seed** is the initial seed for the random number generator used for sampling the input data set.

4. Specify a value for the sample size that is used by performance definitions for models that have an interval target. The default value is **1000**. The **sample size for models with an interval target** is the number of observations from the input data set.

5. Specify a value for the sample size that is used by performance definitions for characteristic and stability analysis. The default value is **10000**. The **sample size for characteristic and stability analysis** is the number of observations from the input data set.

6. Select **Yes** or **No** to specify whether to use the temporary tables on the High-Performance Analytics appliance for performance monitoring. The default is **Yes**.

7. Specify a value for the correlation coefficient (rho) that is used by the binomial test to estimate the probability of default (PD) levels. The default value is **0.04**.

Publish Scoring Options

The **Publish Scoring Options** setting enables you to indicate that the metadata tables be populated in the target database when publishing a scoring function. The default is **Yes**. During the installation and configuration process of the database, the metadata tables must be created in the database if this setting is set to **Yes**. If you plan to use only the SAS Embedded Process publish method to publish scoring model files, this setting can be ignored, and you do not need to create the metadata tables during the database configuration process.

For information about the database configurations, see "Preparing a Data Management System for Use with SAS Model Manager" in *SAS In-Database Products:*

Administrator's Guide. For more information about publishing models to a database, see "Publishing Models to a Database or Hadoop" in *SAS Decision Manager: User's Guide*.

Debug Options

The **Debug Options** setting enables you to use the debug options when executing SAS code within SAS Decision Manager. The default value is `No`.

However, the **Debug Options** setting does not work for scoring tests, performance definitions, and model retrain definitions. To enable debug options with scoring tests, you must add the following line of code to the `appserver_autoexec_usermods.cfg` file in the *****sasconfigdir***Lev#\\SASApp\\WorkspaceServer** directory:

```
options mprint symbolgen notes;
```

Note: You can also add this code using the Edit Start-up Code feature in SAS Decision Manager.

Valid Variable Name Options

The VALIDVARNAME= system option specifies the rules for valid SAS variable names that can be created and processed during a SAS session. The **Valid Variable Name Options** setting enables you to set the VALIDVARNAME system option to ANY when executing SAS code. To do this, you select **Yes**. The default value is **No**.

However, the **Valid Variable Name Options** setting does not work for scoring tests, performance monitoring, or retraining jobs. To use the VALIDVARNAME system option with these features, you must add the following line of code to the `sasv9_usermods.cfg` file in the *****sasconfigdir***Lev#\\SASApp\\WorkspaceServer** directory:

```
options validvarname='any';
```

Note: You can also add this code using the Edit Start-up Code feature in SAS Decision Manager. For more information, see "Edit Start-Up Code" in *SAS Decision Manager: User's Guide*.

When the VALIDVARNAME system option is set to V7, variable names with letters of the Latin alphabet, numerals, or underscores are valid. If you use any other characters, then you must express the variable name as a name literal, and you must also set VALIDVARNAME=ANY. If the name includes either the percent sign (%) or the ampersand (&), then you must use single quotation marks in the name literal in order to avoid interaction with the SAS Macro Facility. For more information, see "VALIDVARNAME= System Option" in *SAS System Options: Reference*.

Note: You must have set the VALIDVARNAME system option to ANY in order to use PMML models that were created using SAS Enterprise Miner in scoring tests, performance monitoring, and reporting within SAS Decision Manager.

In-Database Options

The **In-Database Options** settings enables you to specify the publish type, database connection settings, and publish settings that are used when publishing models to a database using SAS Decision Manager.

To modify the settings for the in-database options:

1. Select a method to publish models to the database for scoring. The default publish type is the **SAS Embedded Process** publish method.

2. Select a database type.

3. Specify values for database settings that are required to publish to the selected database type.

   Here are the available database settings according to the publish method and database type:

Database Settings	SAS Embedded Process	Scoring Function
Server name	• Teradata • Oracle • Netezza • Greenplum • DB2 • Hadoop • SAP HANA	• Teradata • Netezza • Greenplum • DB2
HDFS directory path	Hadoop	Not applicable
Database name or instance number	• Teradata • Oracle • Netezza • Greenplum • DB2 • SAP HANA	• Teradata • Netezza • Greenplum • DB2
User ID	• Teradata • Oracle • Netezza • Greenplum • DB2 • Hadoop • SAP HANA	• Teradata • Netezza • Greenplum • DB2
Server user ID	Not applicable	DB2
Schema	• Oracle • Greenplum • DB2 • SAP HANA	• Greenplum • DB2

4. Specify to use the model manager table when publishing. The default value is **No**. When you are publishing the scoring model files to a database using the SAS Embedded Process publish method, the files are by default stored in the table sas_model_table. If the **Use model manager table** value is set to **Yes**, the scoring

model files are stored in the table sas_mdlmgr_ep. These tables are located in the target database. This setting enables users to separate the SAS Decision Manager scoring model files from the SAS model scoring files when using the SAS Embedded Process publish method.

Note: The sas_mdlmgr_ep table must be created before setting this option to **Yes**. For more information, see "Configuring a Database" in *SAS In-Database Products: Administrator's Guide*.

5. Specify to force the republish of model scoring files by default when using the SAS Embedded Process publish type. The default value is **No**. If you set this setting to **Yes**, then the **Replace scoring files that have the same publish name** option in the Publish Models window in SAS Decision Manager is selected by default.

6. Select the default format of the model publish name when using the SAS Embedded Process publish method. The format selected determines the value that appears for the publish name in the Publish Models window in SAS Decision Manager. The scoring function publish method publish name defaults to the model name.

7. Specify a directory path to store the temporary scoring files. If a value is not specified, the SAS work directory is used by default. However, if the directory is not specified and you select the **Display detailed log messages** option when publishing to a database the SASUSER directory is used.

For information about the database configurations, see "Preparing a Data Management System for Use with SAS Model Manager" in *SAS In-Database Products: Administrator's Guide*. For more information about publishing models to a database, see "Publishing Models to a Database or Hadoop" in *SAS Decision Manager: User's Guide*.

Performance Options

The **Performance Options** setting contains the performance parameters for the PERFORMANCE statement to use the SAS High-Performance Analytics procedures. Currently only the Teradata and Greenplum database types support SAS High-Performance Analytics.

The PERFORMANCE statement defines performance parameters for multithreaded and distributed computing, passes variables about the distributed computing environment, and requests detailed results about the performance characteristics of a high-performance analytics procedure.

The following performance options can be specified for the PERFORMANCE statement.

Option	Description	Default Value
Commit	Specifies an integer to request that the High-Performance Analytics procedure writes periodic updates to the SAS log.	10000

Option	Description	Default Value
CPU count	Specifies how many processors the procedure should assume are available on each host in the computing environment. You can enter the value of ACTUAL, or enter an integer between 1 and 256.	ACTUAL
Database server	Specifies the name of the server for the database as defined through the hosts file and as used in the LIBNAME statement.	
Details	Requests a table that shows a timing breakdown of the procedure steps.	No
Timeout	Specifies the time-out in seconds for a High-Performance Analytics procedure to wait for a connection to the appliance and establish a connection back to the client.	120
Host name	Specifies the name of the appliance. If a value for the Host option is specified, it overrides the value of the grid host environment variable.	
Installation directory	Specifies the name of the directory in which the High-Performance Analytics shared libraries are installed on the appliance.	
Install location	Specifies the name of the directory in which the High-Performance Analytics shared libraries are installed on the appliance. If a value is specified for the Installation directory option, it overrides this option.	

Option	Description	Default Value
Number of nodes	Specifies the number of nodes in the distributed computing environment, provided that the data is not processed alongside the database. You can enter an integer or you can specify a value of ALL if you want to use all available nodes on the appliance without oversubscribing the system.	
Number of threads	Specifies the number of threads for analytic computations. This option overrides the SAS system option THREADS | NOTHREADS. If you do not specify a value for this option, the number of threads are determined based on the number of CPUs on the host on which the analytic computations execute.	
Grid host	Specifies the host name for the grid. If a value for the Host option is specified, it overrides the value of the grid host environment variable.	
Grid RSH command	Specifies the command to run a remote shell.	
Grid reply host	The host name of the client node to which the grid connects.	
Grid port range	Specifies the range of parts that are permitted by the firewall.	
Grid path	Specifies the local directory path for the grid node.	
Grid mode	Specifies whether the HPFORECAST procedure runs in symmetric (SYM) mode or asymmetric (ASYM) mode. The default is symmetric (GRIDMODE=SYM).	Symmetric

For more information about High-Performance Analytics, see *SAS High-Performance Analytics Server: User's Guide*.

Factory Miner Options

If SAS Factory Miner and SAS Decision Manager are licensed at your site, you can register projects segments and models from SAS Factory Miner to the SAS Model Manager model repository. The project segments and models are created within a portfolio. Use the **Automatically set champion** setting to automatically set the project champion in the model repository when a model is set as the champion for a project segment in SAS Factory Miner. The default value is **No**. For information about integrating SAS Factory Miner with SAS Decision Manager, see *SAS Factory Miner: Administration and Configuration*.

Configuring the Limitation for the Number of Observations for a Scoring Result Set

When a scoring test is added on the **Scoring** page of a project in SAS Decision Manager, and the **Type** field is set to **Test**, you can use SAS Management Console to limit the number of observations that a scoring result set can contain.

To configure a limitation for the number of observations:

1. From SAS Management Console, expand the **Application Management** node on the **Plug-ins** tab.

2. Select and expand **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

3. Right-click **Model Manager JavaSvcs 14.1** and select **Properties**.

4. Click the **Advanced** tab. Change the property value for **App.TableObsLimitation** to limit the number of observations in the scoring result set. The default value of **0** indicates that there is no limit to the number of observations that a scoring result set can contain.

5. Click **OK**. The value that you specified now appears in the Number of Observations result set property when you create a scoring test using SAS Decision Manager.

Configure the Limit for Character Strings

Variables that are used in decisions and that are defined as **String** variables are limited to 100 characters unless you define the property **dcb.testing.char.length.override**.

All business rules action terms that are defined as **String** variables and that are not mapped to input variables are limited to 100 characters unless you define the property **brm.testing.char.length.override** in SAS Management Console.

To change the limit for character strings that are used as variables in decisions, complete the following steps in SAS Management Console:

1. On the **Plug-ins** tab, select **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

2. Right-click **Decision Manager 3.1** and select **Properties**. The **Decision Manager 3.1 Properties** dialog box appears.

3. Click the **Advanced** tab, and then click **Add**. The Define New Property dialog box appears.

4. Enter `dcb.testing.char.length.override` for the **Property Name**, enter the string length in the **Property Value** field, and click **OK**.

To change the limit for character strings that are used as action terms in business rules, complete the following steps in SAS Management Console:

1. On the **Plug-ins** tab, select **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

2. Right-click **Business Rules Manager Web 3.1** and select **Properties**. The **Business Rules Manager Web 3.1 Properties** dialog box appears.

3. Click the **Advanced** tab, and then click **Add**. The Define New Property dialog box appears.

4. Enter `brm.testing.char.length.override` for the **Property Name**, enter the string length in the **Property Value** field, and click **OK**.

Review Business Rules Manager Web Properties

Review the Business Rules Manager Web properties in SAS Management Console to ensure that the values are appropriate for your environment. Complete the following steps in SAS Management Console:

1. On the **Plug-ins** tab, select **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

2. Right-click **Business Rules Manager Web 3.1** and select **Properties**. The Business Rules Manager Web 3.1 Properties dialog box appears.

   Note: If you are migrating to SAS Decision Manager 3.1, SAS Management Console displays the release number that you are migrating from. For example, if you are migrating from SAS Decision Manager 2.2, SAS Management Console displays **Business Rules Manager 2.2**.

3. Click the **Settings** tab.

   Review the following properties:

   Location of Code generation macro
   the location of the macro that generates the SAS code for rule sets and rule flows. This property is used only by rule flows that were published using the first maintenance release of SAS Business Rules Manager 1.2. It is not used by rule flows published with later versions.

   Max row count per table
   the maximum number of rows per rule flow test table. This property applies only to the first maintenance release of SAS Business Rules Manager 1.2 and to SAS Business Rules Manager 2.1 and SAS Decision Manager 2.1. It is not used by later versions.

Temporary test code generation directory
applies only to SAS Business Rules Manager 1.2 and the first maintenance release of SAS Business Rules Manager 1.2. This directory is not used by later versions.

Largest allowed uploaded lookup table row count
the maximum number of rows that can be uploaded for a lookup table. The default is 5000. This value should not exceed 10,000.

Maximum Testing Log Length (in lines) of SAS log displayed within User Interface
the maximum number of lines from the SAS log that are displayed on the SAS log section on the **Results** tab for the rule flow.

Support macros in rule expressions
determines whether macros are allowed in rule expressions. Macros are not supported in decision flows that are deployed by SAS Real-Time Decision Manager.

Temporary Location used in Rule Generation
a temporary directory that SAS Decision Manager uses while it generates the SAS code for vocabularies, rule sets, and rule flows. A best practice is to specify a directory that is outside of the SAS configuration directory.

Test Library Root File System Directory
the directory where rule flow tests and test results are saved. As users create additional rule flow tests, administrators might need to delete old test results, or ask users to delete old test cases. A best practice is to specify a directory that is outside of the SAS configuration directory.

Note: If you are running tests that use data sources that are in a distributed environment such as the Hadoop Distributed File System (HDFS), ensure that the SAS Server file system has enough space to accommodate output data in the form of SAS data sets. If sufficient space is not available, you might need to create subsets of the input data to use for testing.

Test Metadata Library Root Directory
the folder in which metadata for rule flow tests is stored.

4. Click the **Advanced** tab.

   Review the following properties:

brm.csvfile.separator
the character that is used as a separator in CSV files that are used to create lookup tables through the SAS Decision Manager interface.

brm.runtime.codetype
determines whether SAS Decision Manager generates DS2 code or DATA step (DS1) code for rule flows. In many cases, you will get better performance by specifying DS1. However, consider specifying DS2 if your input data is in Teradata, Greenplum, or Haddop, and you have installed the SAS Code Accelerator. In addition, the LIKE operator is not supported for DS1.

Note: This property is not predefined. You can add it as a new property.

brm.default.application.server.context.override
the default application server for rule flow testing. You can specify the short logical name such as SASApp. If a value is not specified for this property, SAS Decision Manager looks for a server context named SASApp. If that server does not exist, SAS Decision Manager uses the last one returned from the metadata server.

> *Note:* This property is not predefined. You can add it as a new property.

5. Click **OK** to close the Business Rules Manager Web 3.1 Properties dialog box.

Modify Log File Settings

Log4j Configuration File

SAS Decision Manager uses log4j to perform logging. When SAS Decision Manager starts, the log4j configuration files for the application are read from `SAS-config-dir\Lev1\Web\Common\LogConfig\`. The filenames are `SASBusinessRulesManagerWeb-log4j.xml` and `SASModelManager-log4j.xml`. These files are standard log4j configuration files.

You should not change the existing categories or root logger in a configuration file unless you are instructed to do so by SAS Technical Support.

Logging Priority Levels

You can change the logging priority levels in a log configuration file if needed.

Table 3.1   Logging Priority Levels

Priority	Description
DEBUG	The most verbose logging level. This level displays information that is most useful for debugging an application. SAS Decision Manager should run under this priority only for capturing additional log information. This priority level is not an acceptable priority level for the day-to-day operation of SAS Decision Manager.
INFO	Verbose logging level. This level displays messages that highlight the progress of an application. SAS Decision Manager should run under this priority only for capturing additional log information. This priority level is not an acceptable priority level for the day-to-day operation of SAS Decision Manager.
WARN	Restrictive logging. This level displays information about potentially harmful situations and is an acceptable priority for the day-to-day operation of SAS Decision Manager.
ERROR	The most restrictive logging level. This level displays error events and is an acceptable priority for the day-to-day operation of SAS Decision Manager.

Log Files

SAS Decision Manager writes information to the following log files:

SASBusinessRulesManagerWeb3.1.log
     contains messages from business rules activities

SASModelManager14.1.log
   contains messages from model management activities

SASDecMgrCommon3.1.log
   contains messages from the Workflow and Data plug-ins

SASDecMgrShell3.1.log
   contains general messages from the Shell

By default, the application writes the SAS Decision Manager log files to
`SAS-config-dir\Lev1\Web\Logs\SASServer7_1\`. The
SASModelManager14.1.log file is written to
`SAS-config-dir\Lev1\Web\Logs\SASServer11_1\`. You can change the
location of these log files in the configuration file. Changes to the configuration file take
effect when the middle-tier application server is restarted. See "Administering Logging
for SAS Web Applications" in *SAS Intelligence Platform: Middle-Tier Administration
Guide* for more information about this configuration file.

SAS Decision Manager creates new log files each day. For information about logging
configurations, see "Modifying Your Server Logging Configurations" in *SAS
Intelligence Platform: System Administration Guide*.

Turn on SQL Logging

To turn on SQL logging and write SQL parameter values for each query to the log file,
add the following categories to the log4j.xml configuration file:

```
<category additivity="false" name="org.hibernate.type">
   <priority value="TRACE"/>
   <appender-ref ref="SAS_FILE"/>
</category>


<category additivity="false" name="org.hibernate.SQL">
   <priority value="DEBUG"/>
   <appender-ref ref="SAS_FILE"/>
</category>
```

Manage Directories for Business Rules Content

SAS Decision Manager creates two directories for business rules metadata: **`Products`**
and **`/System`**.

SAS Decision Manager creates a location for published XML files,
`sasdav/Products`. The BusinessRuleFlow metadata object does not delete the XML
documents stored in this location in order to ensure that an audit trail is maintained.

Before you delete any XML content from **`sasdav/Products`**, you should do the
following:

1. Back up all versions of the content. The easiest way to back up the content is to use
   SAS Management Console to export the BusinessRuleFlow object that refers to the
   content.

2. Ensure that no BusinessRuleFlow objects refer to the content.

Configure Users Authenticated by Kerberos

To use Kerberos authentication, each SAS Decision Manager user must have a valid Kerberos ticket to access SAS Decision Manager. However, users who are authenticated by Kerberos cannot write the publish results files to the SAS Content Server when publishing a model because they have not supplied a password to SAS Model Manager. Therefore, additional post-installation configuration steps are needed so that users can publish models to a database or a Hadoop Distributed File System (HDFS) from SAS Decision Manager:

1. Create an internal user, such as **sascs@saspw**, and add the user to the Model Manager Advanced Users group. For more information, see Chapter 6, "Configuring Users, Groups, and Roles," on page 57.

2. Create an operating system group, if one has not been previously created. For more information, see "Creating Operating System Accounts for Product Administrators and Users" on page 13.

3. Ensure that each user who publishes models is part of the operating system group that you created. Users in the Model Manager Advanced Users and Model Manager Administrator Users groups can publish models. Therefore, it is recommended that these users be included in the operating system group.

4. As an operating system administrator, perform the following tasks:

   a. Create a directory and add Read permissions for the operating system group that you created. Only the operating system administrator should have Write permissions.

   b. Create an encoded password file using the PWENCODE procedure.

   LAX example:

   ```
   filename pwfile "<directory-path>/sascs.pwd";
   proc pwencode in="<internal-user-password>" out=pwfile;
   run;
   filename pwfile; /*closes the file*/
   ```

   c. Move the file into the directory that you created and set the permissions for the file. Set Read permissions for the operating system group that you created and Write permissions for the operating system administrator.

   LAX example:

   ```
   sudo mv "<directory-path>/sascs.pwd";
   sudo chgrp mdlmgr /sascs/sascs.pwd
   sudo chown root /sascs/sascs.pwd
   sudo chmod 750 /sascs/sascs.pwd
   ```

5. After you have completed the rest of the required post-installation configuration and verification steps, sign in to SAS Decision Manager. Then edit the start-up code to specify to use the internal user for publishing to Hadoop.

   To edit the start-up code:

   a. Select **Actions** ⇨ **Edit Start-up Code**. The Edit Start-up Code window appears.

   b. Enter the SAS code.

LAX example:

```
filename pwfile "<directory-path>/sascs.pwd";
data _null_;
infile pwfile obs=1 length=l;
input @;
input @1 line $varying1024. l;
call symput('_MM_Password',substr(line,1,l));
run;
filename pwfile; /*closes the file*/
%let _MM_User=sascs@saspw;
```

c. Click **Run Now**.

d. Click the **Log** tab to see the SAS log. Ensure that there are no errors in the log.

e. Click **OK**. The SAS code is saved in the Edit Start-up Code window.

> *Note:* If you save the code without running it (by clicking **OK**), the code is automatically added to the header of the PublishModel.sas code that is created when models are published.

Note: The internal user should be used only to publish models from SAS Decision Manager in an environment that is authenticated by Kerberos, and should not be used to perform any other tasks. Limit access to the password file as few users as possible.

Chapter 4
Performing Upgrade Tasks

About the Upgrade Process

You can upgrade from SAS Decision Manager 2.2 to SAS Decision Manager 3.1.

When you are upgrading to SAS Decision Manager 3.1 on the third maintenance release of SAS 9.4, you must complete additional installation and configuration steps. SAS Decision Manager 3.1 includes a new SAS Decision Manager Common component called Decision Builder and is integrated with SAS Lineage. Therefore, two passes of the SAS Deployment Wizard installation and configuration process are required in order to complete the upgrade. During the first pass of the SAS Deployment Wizard, it upgrades your existing deployment. After the upgrade has completed successfully, the configuration stage begins. After the configuration stage has completed, you must run the SAS Deployment Wizard again to install the new product components and to complete the configuration.

The SAS Decision Manager upgrade process supports upgrading to a database from the same vendor as the database that you are currently using. Upgrading from a database based on Oracle to a database based on PostgreSQL, or vice versa, is not supported.

If you are using Oracle for your SAS Decision Manager database, the upgrade process assumes that the upgraded environment uses the same instance of Oracle. The upgrade process does not support moving to a different Oracle database server.

Pre-upgrade Steps

Before you upgrade to SAS Decision Manager 3.1, record the database settings in your current environment. You must enter this information in SAS Deployment Wizard.

- If you are upgrading from SAS Decision Manager 2.2, record the database name and the user ID for your SAS Decision Manager Common Data Server database. The default database name is **dcmdb**.

  You can find the database name in the **SASCONFIG/Web/WebAppServer/ SASServer7_1/conf/server.xml** file on the middle-tier server. Find the resource with the name **sas/jdbc/DecisionManagerDS**, and look for the value of the **url** attribute. The database name is the text after the final forward slash (/) in the URL. For example, if the attribute is **url="jdbc:postgresql://*host*: 10482/dcmdb"**, then the database name is **dcmdb**.

- If you are using Oracle for your database, ensure that the Oracle client is installed on your server tier and that there is a matching tnsnames.ora file that corresponds to your database. Record the information in the following table.

  You enter this information in the SAS Decision Manager Database Properties and SAS Decision Manager Database JDBC Properties windows.

Table 4.1 *SAS Deployment Wizard Information for Oracle*

Prompt	Description
Host Name	Specifies the fully qualified host name of the server on which the database is installed.
Port	Specifies the port number that is used by the database. The default port for Oracle is 1521.
Directory containing JDBC driver jars	Specifies the location of the database vendor's JDBC JAR file. This file must be available on the middle tier and on any machine on which you are deploying SAS Decision Manager in order to configure SAS Decision Manager database.
	See "Verify JDBC Drivers for Oracle" on page 5 for more information.

Prompt	Description
Database SID or Service Name	Specifies the Oracle database name. The database name must match either the service name or the Oracle site identifier (SID), both of which can be found in the tnsnames.ora file.
	If you select **Use Oracle database name as a Service Name**, then you must enter the service name that is specified in the tnsnames.ora file. For example, if you had the following entry in the tnsnames.ora file, you would enter **monitordb** in the **Database SID or Service Name** field:

```
monitordb =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = TCP_COMM)
          (PROTOCOL = TCP)
          (HOST = hostname.your.company.com)
          (PORT = 1521)
        )
    )
    (CONNECT_DATA =
      (SERVICE_NAME = monitordb)
    )
  )
```

	Note: In the tnsnames.ora file, the **Net Service Name** and the **Service Name** fields must be the same.
	You can also find the Oracle SID in the tnsnames.ora file. Alternatively, you can run the following query using a database user ID on your Oracle instance:
	select instance from v$thread
User ID	Specifies the user ID of the database user whose credentials are used to access SAS Decision Manager data on the server.
Password	Specifies the password of the user ID whose credentials are used to access SAS Decision Manager data on the server.
Schema Pattern	Specifies the schema name for the database. The default schema is the same as the user ID.

Run the SAS Deployment Wizard

Follow the instructions provided in "Add SAS Products That Require Configuration" in *SAS Intelligence Platform: Installation and Configuration Guide* in order to complete the installation and configuration process for an upgrade.

When you run the SAS Deployment Wizard the second time, the following products are installed:

- SAS Lineage Mid-Tier

- SAS Micro Analytic Service Java Interfaces

- SAS Micro Analytic Service Rest API

Note: In a multi-machine environment, the server products are typically installed on the SAS Application Server. The rest of the products are on the SAS Middle-Tier Server.

Note: If you are configuring multiple Web Application servers, **SASServer13** is added to the deployment for the additional products, so be aware of the need for additional ports.

When you run the SAS Deployment Wizard the second time, verify the following:

- Verify that you are using the new SAS installation data (SID) file. The SID file for your software (your order) is located in the **sid_files** directory at the root of your SAS Software Depot or, if you have received media, on the first disk of that installation media.

- Verify that your plan file contains SAS Model Manager and SAS Decision Manager Common products. There is not a standard plan for SAS Decision Manager, so you need a custom plan.

- Before the configuration stage begins, make sure that the following SAS Services are started:

  - SAS Metadata Server

  - SAS Web Infrastructure Platform Data Server

  - SAS Decision Manager Common Data Server (if you are not using Oracle for your database)

  - SAS Web Server (httpd - WebServer)

  - SAS Object Spawner

  - SAS JMS Broker

  - SAS Cache Locator

- Verify that the following products are selected for configuration:

  - SAS Web Application Server Configuration

  - SAS Lineage Mid-Tier

  - SAS Help Viewer for Midtier Applications

  - SAS Micro Analytic Service Rest API

Post-upgrade Steps

1. After you complete the upgrade process with the SAS Deployment Wizard, see the Instructions.html file. The Instructions.html file is located in **\sasconfigdir \Lev#\Documents**. The default URL is **http://*host_name:port*/ SASDecisionManager**. The default URL for Decision Builder is **http:// *host_name:port*/SASDecisionManager/DecisionBuilder**.

2. (Optional) Configure your deployment to use HTTPS. See "Configure Your Deployment for HTTPS" on page 22 for more information.

3. Update your user group memberships, authorization, roles, and capabilities as needed. Make sure that you create the Decision Builder User role and add it to the Decision Manager Users group. See "Updating Groups and Roles for SAS Decision Manager 3.1" on page 66 for more information. See "Using the Decision Builder" in *SAS Decision Manager: User's Guide* for information about accessing and using Decision Builder.

4. (Optional) Add the Visual Analytics: Data Building and Data Management: Lineage roles to the Decision Manager Users group. See "Administering Group and Role Membership" on page 63 for more information. These roles enable users to run SAS Visual Data Builder and view lineage information for rule flows and models.

5. Grant users permissions to the Decision Builder testing directory.

6. Review the configuration properties in SAS Management Console. Ensure that the values are appropriate for the new environment. See "Review Business Rules Manager Web Properties" on page 34 for more information.

Chapter 5
Performing Migration Tasks

About the Migration Process

You can migrate from SAS Decision Manager 2.2 and 3.1 to SAS Decision Manager 3.1.

The SAS Decision Manager migration process supports migrating to a database from the same vendor as the database that you are currently using. Migrating from a database based on Oracle to a database based on PostgreSQL, or vice versa, is not supported.

If you are using Oracle for your database, the migration process assumes that the migrated environment uses the same instance of Oracle. The migration process does not support moving to a different Oracle database server.

Pre-migration Steps

For information about the pre-migration tasks that you must perform, see "Performing Pre-migration Tasks" in *SAS Intelligence Platform: Migration Guide*. Here are some important steps to help with your migration:

- Back up your SAS system, including servers and desktop clients.

- Back up the SAS Web Infrastructure Platform Services database if you are migrating a SAS 9.3 or SAS 9.4 system.

- Back up the SAS Model Manager database if you are migrating SAS Model Manager 12.1 on a SAS 9.3 system or SAS Model Manager 12.3 on a SAS 9.4 system.

- Back up the SAS Decision Manager database if you are migrating from SAS Decision Manager 2.2 or 3.1 to SAS Decision Manager 3.1 on a SAS 9.4 system.

- If you are moving to a new system, ensure that the required operating system user accounts that you use for SAS in your current operating system also exist in your new operating system. These accounts are required for running scoring tests, performance definitions, retraining models, and scheduling jobs.

- Before you migrate to SAS Decision Manager 3.1, you should record the database settings in your current environment. Verify that these settings are entered in the SAS Deployment Wizard when you run the migration.

  - If you are migrating from SAS Model Manager 12.1, or 12.3, record the database name and the user name for your SAS Model Manager database. The default database name is `mdlmgrdb`.

  - If you are migrating from SAS Decision Manager 2.2 or later, record the database name and the user ID for your SAS Decision Manager Common Data Server database. The default database name is `dcmdb`.

    You can find the database name in the `SASCONFIG/Web/WebAppServer/SASServer7_1/conf/server.xml` file on the middle-tier server. Find the resource with the name `sas/jdbc/DecisionManagerDS`, and look for the value of the `url` attribute. The database name is the text after the final forward slash (/) in the URL. For example, if the attribute is `url="jdbc:postgresql://host:10482/dcmdb"`, then the database name is `dcmdb`.

  - If you are using Oracle for your database, ensure that the Oracle client is installed on your server tier and that there is a matching tnsnames.ora file that corresponds to your database.

  - If you are using Oracle for your database, record the information that is listed in the following table.

    You enter this information in the SAS Decision Manager Database Properties and SAS Decision Manager Database JDBC Properties windows.

Table 5.1 SAS Deployment Wizard Information for Oracle

Prompt	Description
Host Name	Specifies the fully qualified host name of the server on which the database is installed.
Port	Specifies the port number that is used by the database. The default port for Oracle is 1521.
Directory containing JDBC driver jars	Specifies the location of the database vendor's JDBC JAR file. This file must be available on the middle tier and on any machine on which you are deploying SAS Decision Manager in order to configure SAS Decision Manager database. See "Verify JDBC Drivers for Oracle" on page 5 for more information.

Prompt	Description
Database SID or Service Name	Specifies the Oracle database name. The database name must match either the service name or the Oracle site identifier (SID), both of which can be found in the tnsnames.ora file.
	If you select **Use Oracle database name as a Service Name**, then you must enter the service name that is specified in the tnsnames.ora file. For example, if you had the following entry in the tnsnames.ora file, you would enter **monitordb** in the **Database SID or Service Name** field:

```
monitordb =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS =
        (COMMUNITY = TCP_COMM)
        (PROTOCOL = TCP)
        (HOST = hostname.your.company.com)
        (PORT = 1521)
      )
    )
    (CONNECT_DATA =
      (SERVICE_NAME = monitordb)
    )
  )
```

	Note: In the tnsnames.ora file, the **Net Service Name** and the **Service Name** fields must be the same.
	You can also find the Oracle SID in the tnsnames.ora file. Alternatively, you can run the following query using a database user ID on your Oracle instance:
	select instance from v$thread
User ID	Specifies the user ID of the database user whose credentials are used to access SAS Decision Manager data on the server.
Password	Specifies the password of the user ID whose credentials are used to access SAS Decision Manager data on the server.
Schema Pattern	Specifies the schema name for the database. The default schema is the same as the user ID.

- If you are using Oracle for your database, verify that you have the correct JDBC drivers. For more information, see "Verify JDBC Drivers for Oracle" on page 5.

- Use the SAS Migration Utility to create a migration package. You must enter the configuration properties for the SAS Model Manager Mid-Tier (MMAPI) database management system in the SAS Migration Utility properties file.

  The values for the SAS Migration Utility properties can be found in SAS Management Console. Select the **Folders** tab and expand **System** ⇨ **Applications** ⇨ **SAS Model Manager Mid-Tier**. Select the **Model Manager Mid-Tier** *version* folder, right-click the **Model Manager-Mid-Tier** *version* application object, and then select **Properties** ⇨ **Configuration**. The prefix for the configuration properties that are equivalent to the SAS Migration Utility properties is dbms.mmapi.

  Note: If you are migrating from SAS Model Manager 12.1 or 12.3, the default name of the SAS Model Manager database is **mdlmgrdb**. Beginning with SAS Model Manager 13.1, the database is called the SAS Decision Manager database. The default name is **dcmdb**.

Here is the list of configuration properties that should be used to populate the migration properties:

SAS Migration Utility Properties	Configuration Properties in SAS Management Console
mmapi.data.dbms.type	dbms.mmapi.type
mmapi.dbms.data.name	dbms.mmapi.name
SMU.mmapi.dbms.host	dbms.mmapi.host
SMU.mmapi.dbms.port	dbms.mmapi.port
SMU.mmapi.dbms.userid	dbms.mmapi.userid
SMU.mmapi.dbms.password	<not stored here>
SMU.mmapi.jdbc.driver	dbms.mmapi.jdbc.dir

For more information, see "SAS Migration Utility Reference" in *SAS Intelligence Platform: Migration Guide*, and "SAS Model Manager Properties" in *SAS Intelligence Platform: Migration Guide*.

Post-migration Steps

Overview of Post-migration Steps

After you have migrated to SAS Decision Manager 3.1 on SAS 9.4, perform the following post-migration steps:

1. Copy or move the contents of the rule flow testing directories. See "Copying Rule Flow Test Results" on page 51 for more information.

2. Copy or move the contents of the channels directory from the source machine to the target machine. See "Copying Publishing Channels" on page 51 for more information.

3. Run post-migration macros if you are migrating from one UNIX operating system to a different type of UNIX system. See "Migrate Model Information to a New Operating System" on page 54 for more information.

4. Run the appropriate migration script to migrate your database to SAS Decision Manager 3.1. See the following topics for additional information:

   • "Migrate from Version 2.x to 3.1 for Oracle" on page 52

   • "Migrate to Version 3.1 for SAS Decision Manager Common Data Server (Single Tier)" on page 52

   • "Migrate to Version 3.1 for SAS Decision Manager Common Data Server (Multiple Tiers)" on page 54

Note: If you are migrating from SAS Decision Manager 3.1 to 3.1 (hardware upgrade) and you are using Oracle for your database, you do not need to run a migration script.

5. (Optional) Configure your deployment to use HTTPS. See "Configure Your Deployment for HTTPS" on page 22 for more information.

6. Update your user group memberships, authorization, roles, and capabilities as needed. See "Updating Groups and Roles for SAS Decision Manager 3.1" on page 66 for more information.

7. (Optional) Add the Visual Analytics: Data Building and Data Management: Lineage roles to the Decision Manager Users group. See "Administering Group and Role Membership" on page 63 for more information. These roles enable users to run SAS Visual Data Builder and view lineage information for rule flows.

8. Review the configuration properties in SAS Management Console. Ensure that the values are appropriate for the new environment. See "Review Business Rules Manager Web Properties" on page 34 for more information.

9. If you are migrating from SAS Model Manager 12.1 or earlier, see *SAS Model Manager: Migration Guide* for additional post-migration steps that you might need to perform.

Note: All user-defined model templates are copied from the source system to the target system by the migration process. This content includes all user-defined life cycle templates, model templates, report templates, and SAS code files. The user-defined templates are located on the SAS Content Server at http://*hostname*:*port*/ SASContentServer/repository/default/ModelManager/ConfigTemplates/ext/.

Copying Rule Flow Test Results

Rule flow testing results are not automatically migrated. For any results that you want to access with SAS Decision Manager 3.1:

1. Copy the test results from your old deployment to the new deployment. The directory path for rule flow test results is specified by the Business Rules Manager Web 3.1 property **Test Library Root File System Directory** in SAS Management Console. You must use the same directory path in the new deployment. See "Review Business Rules Manager Web Properties" on page 34 for more information.

2. (Optional) If you are running in a UNIX operating environment and the user that needs to access the test results is different from the current user, use the `chown -R` command to change the ownership of the testing folders. For more information, refer to the UNIX man page for the `chown` command.

3. Update the relationship information for the test results so that users can view the test results and re-run the test cases. To update the relationship information, run the following REST service:

   `http://host:port/SASBusinessRulesManagerWeb/rest/RuleflowTestLibMigrate`

Copying Publishing Channels

When migrating from a previous release of SAS Decision Manager, you must manually copy the contents of the channels directory from your old system to a directory on the new machine that is running SAS Workspace Server. It is recommended that you use the same directory name when migrating from an existing SAS Decision Manager 3.1

deployment to a new deployment of SAS Decision Manager 3.1. If you use the same directory name, you do not need to copy the contents of the channels directory. The default directory path for channels is ***SASConfigDir*\\Lev#\\AppData \\SASModelManager14.1\\Channels**.

For example, in Windows the path might be **C:\\SAS\\Config\\Lev1\\AppData \\SASModelManager14.1\\Channels**.

If you use a different directory name, then you must modify the channel persistent store directory location in the SAS Management Console. For more information, see "Modify an Existing Channel or Channels Node Location" on page 88.

Migrate from Version 2.x to 3.1 for Oracle

The scripts for migrating an Oracle database are located in the following directory:

***SASHome*/SASDecisionManagerCommonMidTierforDecisionManager/3.1/ Config/Deployment/dbscript/oracle/migration/**.

To migrate an Oracle database from SAS Decision Manager 2.2 to 3.1, run the **migration_brm_*version*_to_brm_3.1.sql** script for your current release of SAS Decision Manager.

For example, you can use SQL*Plus to run the script to migrate from SAS Decision Manager 2.2 to 3.1 as follows:

```
sqlplus username@tnsname @/install/SASHome/
SASDecisionManagerCommonMidTierforDecisionManager/3.1/Config/Deployment/
dbscript/oracle/migration/migration_brm_2.2_to_brm_3.1.sql schemaName
```

Migrate to Version 3.1 for SAS Decision Manager Common Data Server (Single Tier)

If you are migrating from SAS Decision Manager 2.2 or later to 3.1 and you are using the SAS Decision Manager Common Data Server, you must run the database migration script for your operating environment.

1. Verify that the SAS Decision Manager Common Data Server is running on both the source and target machines.

2. Shut down all SAS Web Application Server processes.

3. Run the database migration script for your operating environment. The script for Windows platforms is named **postgres-migration.bat**, and the script for UNIX platforms is named **postgres-migration.sh**. They are located in the following directory:

   ***SASHome*/ SASDecisionManagerCommonMidTierforDecisionManager/3.1/ Config/Deployment/Migration/**

   Note: This directory contains a README.TXT file that contains information about the parameters for these scripts.

   When you run the migration script, substitute the correct values for the release you are migrating from, the server name, port number, and user ID for your database. The script prompts you to enter a password. The syntax for these scripts is as follows:

   postgres-migration.[bat | sh] *SASHome version source_port source_host source_user source_db_name target_port target_host target_admin target_user target_db_name*

<target_exists>

SASHome
> specifies the SAS Home installation directory location. In Windows operating environments, it is recommended that you enclose this parameter in double quotation marks.

version
> specifies the version number of your current (source) database from which you are migrating content. Specify **2.2** or **3.1**.

source_port
> specifies the port number of the database from which you are migrating content.

source_host
> specifies the host name of the database from which you are migrating content.

source_user
> specify the user ID for Decision Manager Common Middle Tier. This value must be a user ID that has access to all of the database content that needs to be migrated. You can find the correct value for this parameter in the SASCONFIG/Web/WebAppServer/SASServer7_1/conf/server.xml file on the middle-tier server for the source system. Specify the value of the **user** attribute of the resource with name **sas/jdbc/DecisionManagerDS**.

source_db_name
> specifies the name of the database from which you are migrating content. You can find the correct value for this parameter in the SASCONFIG/Web/WebAppServer/SASServer7_1/conf/server.xml file on the middle-tier server for the source system. Find the resource with the name **sas/jdbc/DecisionManagerDS**, and look for the value of the **url** attribute. The database name is the text after the final forward slash (/) in the URL. For example, if the attribute is **url="jdbc:postgresql://*host*:10482/dcmdb"**, then specify **dcmdb** for *source_db_name*.

target_port
> specifies the port number of the database to which you are migrating the content.

target_host
> specifies the host name of the database to which you are migrating the content.

target_admin
> specifies the user ID of the database administrator for the database to which you are migrating the content. This user ID is used to clean the target database and prepare it for the migrated content.

target_user
> specifies a database user ID for the database to which you are migrating the content. This user ID is assigned ownership of the migrated content.

target_db_name
> specifies the database name of the database to which you are migrating the content. The default target database name is **dcmdb**.

target_exists
> specifies whether the target database exists. The default value is **YES**. If you have attempted to run this migration script but the migration failed, specify **NO**. Normally, the script creates a backup of the source database. If you specify **NO**, the script does not create an additional backup.
>
> *Note:* This parameter is optional.

Migrate to Version 3.1 for SAS Decision Manager Common Data Server (Multiple Tiers)

If you are migrating from SAS Decision Manager 2.2 or later to 3.1 and you are using the SAS Decision Manager Common Data Server, you must run the database migration script for your operating environment. You must run the post-migration script on the server tier. In a multiple-tier environment, you must copy the script from the middle tier to the server tier before you run it.

1. Verify that the SAS Decision Manager Common Data Server is running on both the source and target machines.

2. Shut down all SAS Web Application Server processes.

3. Create a directory on the server-tier machine (the machine that is running that SAS Decision Manager Common Data Server) for the migration scripts. The directory can reside in the SAS Home installation directory (for example, **SASHome/ DCMDB_Migrate**), but that is not a requirement.

4. Copy the **Config** directory on the middle-tier machine at **SASHome/ SASDecisionManagerCommonMidTierforDecisionManager/3.1/Config** to the directory that you created on the server-tier machine.

5. Edit the database migration script for your platform that is in the directory that you created in Step 3. Change the value that is specified for the DCM_DATA_DIR= parameter to point to the directory that you created in Step 3 (for example, **DCM_DATA_DIR=$SAS_HOME/DCMDB_Migrate**).

   The migration scripts are in the **Config/Deployment/Migration** subdirectory. The script for Windows platforms is named **postgres-migration.bat**, and the script for UNIX platforms is named **postgres-migration.sh**. If you specified **DCMDB_Migrate** for the new directory in Step 3, the script that you need to edit is located in the directory **SASHome/DCMDB_Migrate/Config/Deployment/ Migration/**.

6. Run the migration script on the server-tier machine as described in Step 3 of "Migrate to Version 3.1 for SAS Decision Manager Common Data Server (Single Tier)" on page 52. Substitute the correct values for the release that you are migrating from, the server name, port number, and user ID for your database. The script prompts you to enter passwords for different user accounts. Note the user account that is specified in the prompt to ensure that you enter the correct password.

Migrate Model Information to a New Operating System

If you are migrating from one UNIX operating system to another, then some post-installation steps are required to complete the migration process. SAS Decision Manager provides post-migration macros to assist with this process. Only SAS files on the WebDAV server in the **\\ModelManagerDefaultRepository\Project \Version\Resources** directory are handled by the post-migration macros. If you have SAS files in another directory location, you must manually migrate them.

Note:

- You must have access to the SAS deployment on the source system and target system to run these macros.

- The %MM_migrationStep1() and %MM_migrationStep2() macros are available in the SAS catalog sashelp.modelmgr.migration.source.

Perform the following steps to ensure that all data, content, and link and filename references that are used by SAS Decision Manager are accessible by the new SAS 9.4 deployment:

1. Run the %MM_migrationStep1() macro on page 117 on the SAS Workspace Server in your source system.

2. Run the %MM_migrationStep2() macro on page 118 on the SAS Workspace Server in your target system.

Note: For more information about the migration macros, see Chapter 11, "Post-Migration Macros," on page 117.

Chapter 6

Configuring Users, Groups, and Roles

Security Administration Tasks for SAS Decision Manager

Security administration for SAS Decision Manager consists of the following tasks:

- administering SAS identities for your users by adding account information to the SAS Metadata Server

- administering groups of users in order to simplify the management of roles

- administering roles, which provide users with access to specific application features

The information included here is a brief introduction to the concepts of users, SAS identities, groups, roles, and capabilities. For complete information about security administration, see *SAS Management Console: Guide to Users and Permissions* and *SAS Intelligence Platform: Security Administration Guide*.

Administering SAS Identities for Users

Overview of SAS Identities

For each SAS Decision Manager user, you must create an individual SAS identity on the SAS Metadata Server. The SAS identity is a copy of the ID with which the user logs on to SAS applications. Based on this identity, the system can determine who can access which application and can audit individual actions in the metadata layer. The SAS identity consists of a name and the user ID for the user's external account. This ID can be any type of account that is known to the metadata server's host, such as an LDAP account, Active Directory account, host account, or other type of account.

When you are entering user IDs for Windows accounts, be sure to qualify the ID (for example, `WIN\myID` or `myID@mycompany.com`).

In a Windows environment, add new users to the **Log on as a batch job** local security policy on the machine that hosts the SAS Workspace Server.

Note: Enter a user ID for Microsoft Windows in the format *domain\userID*. In order for users to be able to schedule jobs with SAS Decision Manager, you must specify a valid password in each user's account. Users must sign in with the same user ID (domain\userID) and password that you specified in their user account.

Note: Users who log on to SAS Decision Manager using an internal account (a user ID that ends in @saspw) cannot access all of the features of the application. All users should be assigned external accounts.

The following users are created as part of the SAS Decision Manager installation process:

Table 6.1   Types of Users

User	Description
SAS Administrator	This user has access to all SAS Management Console capabilities and metadata administrative tasks.
SAS Demo User	This user is optional. You can choose to create this user during installation. However, this user is not assigned to a group during installation.

Creating SAS Identities

To create SAS identities for your users, manually enter the information for each user through the User Manager plug-in in SAS Management Console. If you have a large number of users, then you can extract user and group information from one or more enterprise identity sources. You can then use SAS bulk-load macros to create the identity metadata from the extracted information.

For more information about creating and managing identities, see *SAS Management Console: Guide to Users and Permissions*. For information about the SAS bulk-load macros, see "User Import Macros" in *SAS Intelligence Platform: Security Administration Guide*.

Groups and Group Membership

About Groups

A group is a set of users. Groups enable you to grant multiple users membership in a role or permissions to metadata, thus simplifying security administration. You can create as many groups as are needed in order to manage your installation.

TIP A group's membership can include other groups as well as individual users. This enables you to create a nested group structure.

Predefined User Groups in SAS Decision Manager

Table 6.2 *Predefined User Groups*

Group	Description
Public	This group includes everyone who can access the metadata server, either directly or through a trust relationship. If a user is able to log on to a client application but does not have an individual SAS identity, the user is assumed to be in the public group. Because this group has implicit membership, you cannot explicitly add or remove users from this group.
SAS Users	This group includes everyone who can access the metadata server, either directly or through a trust relationship. If a user is able to log on to a client application but does not have an individual SAS identity, the user is assumed to be in the public group. Because this group has implicit membership, you cannot explicitly add or remove users from this group.
SAS Administrators	This is a standard group for metadata administrators. In a standard configuration, members are granted broad access and administrative capabilities, but are not unrestricted.

Group	Description
Decision Manager Common Administrators	This group has administrative permissions. Membership in this group is required to administer workflows.
	In your initial installation, this group is a member of the following roles:
	• Decision Manager Common: Administration
	• Business Rules Manager: All Capabilities
	• Model Manager: Administration Usage
	• Decision Builder User (through the Decision Manager Users group)
Decision Manager Users	This group is created during the installation process. Members of this group have permission to read, add, or delete table summary information in the Data category.
	This group is a member of the Decision Builder User role, so members of this group have permission to read, create, update, delete, run, and publish a decision.
	During configuration, this group was associated with an identity that enables members to access the database during rule flow and table summary execution.
	Note: This group is the only group that is granted permission to publish business rules content to the SAS Content Server by default.
	Note: Unless you make configuration changes, users who do not have administrator permission must be members of this group. If you want these users to be members of a different group, you must grant the group permissions that enable members to do the following:
	• access the database (using the necessary identity)
	• publish content to the folders for SAS Decision Manager on the SAS Content Server
Model Manager Administrator Users	This group has administrative permissions in the Projects and Portfolio categories.
	For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
Model Manager Advanced Users	This group has permissions to read, write, and delete content in the Projects and Portfolios categories.
	For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
Model Manager Users	This group has permission to read content in the Projects category.
	For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
SAS System Services	This group enables members to export files on the **Folders** tab of SAS Management Console.

Roles and Capabilities

About Roles and Capabilities

A role manages the availability of application features such as menu items and plug-ins. An application feature that is under role-based management is called a capability.

Certain actions are available only to users or groups that have a particular role. Any user or group who is a member of a role has all of that role's capabilities.

Roles can contribute to one another. A role automatically includes all of the capabilities of a role that contributes to it.

Roles differ from permissions. In general, roles do not affect access to metadata or data.

Predefined Roles and Capabilities for SAS Decision Manager

Your installation includes several predefined roles for administrators and users of SAS Decision Manager. Depending on what software you have installed, you might have other predefined roles.

Note: The ability to access and update metadata is subject to permissions that are placed on that metadata. These roles do not affect permissions.

Table 6.3   *Predefined User Roles and Capabilities*

Role	Description
Decision Manager Common: Administration	Enables users to perform all Decision Manager Common tasks, including administering workflows.
	This role is assigned to the group Decision Manager Common Administrators and has the Decision Manager Common: Workflow category capability.
Business Rules Manager: All Capabilities	Enables users to create, edit, and delete all business rules content, including vocabularies, entities, terms, lookup tables, rule sets, and rule flows.
Business Rules Manager: Rule Flow and Rule Set Designer	Enables users to create, edit, and delete rule sets and rule flows.
Business Rules Manager: Rule Flow and Rule Set Read-Only	Enables users to view rule sets and rule flows.
Business Rules Manager: Vocabulary and Lookup Designer	Enables users to create, edit, and delete vocabularies, entities, terms, and lookup tables.
Business Rules Manager: Vocabulary and Lookup Read-Only	Enables users to view vocabularies, entities, terms, and lookup tables.

Role	Description
Model Manager: Administration Usage	Enables users to perform all model management tasks. For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
	This role is assigned to the group Model Manager Administrator Users and has the following Model Manager Plug-in capabilities by default:
	• Model Projects category
	• Model Portfolios category
Model Manager: Advanced Usage	Enables users to perform all model management tasks except for tasks that can be performed only by an application administrator. For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
	This role is assigned to the group Model Manager Advanced Users and has the following Model Manager Plug-in capabilities by default:
	• Model Projects category
	• Model Portfolios category
Model Manager: Usage	Enables users to perform general model management tasks, such as viewing project and model information. For specific user tasks that can be performed, see "Model Management User Tasks" on page 67.
	This role is assigned to the group Model Manager Users and has the following Model Manager Plug-in capabilities by default:
	• Model Projects category
Comments: Administrator	Enables users to edit or delete comments.
	The ability to edit and delete comments is controlled by the capabilities under **Applications ⇨ SAS Application Infrastructure ⇨ Comments** in SAS Management Console.
Data Management: Lineage	Provides default access to the SAS Lineage application. This role is predefined, but it is not automatically added to the Decision Manager Users group. To enable SAS Decision Manager users to access SAS Lineage, add the Data Management: Lineage role to the Decision Manager Users group. See "Adjust Group or Role Membership" in *SAS Management Console: Guide to Users and Permissions* for instructions.

Role	Description
Decision Builder User	Enables users to read, create, edit, delete, and publish decisions in Decision Builder. *Note:* This role is predefined when you install a new deployment or migrate to SAS Decision Manager 3.1. If you upgrade to SAS Decision Manager 3.1, you must create this role. See "Create the Decision Builder User Role" on page 66 for more information.
Management Console: Advanced	Provides access to all plug-ins in SAS Management Console. This role is assigned to the group SAS Administrators.
Metadata Server: Operation	Supports adding metadata repositories and operating the metadata server. This role is assigned to the group SAS Administrators.
Metadata Server: User Administration	Supports management of users, groups, and roles other than the unrestricted users role. This role is assigned to the group SAS Administrators.
Metadata Server: Unrestricted	Provides all capabilities in SAS Management Console and provides access to all metadata. This role is assigned to the group SAS Administrator Users.
Visual Analytics: Data Building	Enables users to access SAS Visual Data Builder. This role is predefined, but it is not automatically added to the Decision Manager Users group. To enable SAS Decision Manager users to access SAS Visual Data Builder, add the Visual Analytics: Data Building role to the Decision Manager Users group. See "Adjust Group or Role Membership" in *SAS Management Console: Guide to Users and Permissions* for instructions.

Administering Group and Role Membership

To administer group and role membership, use the User Manager plug-in in SAS Management Console.

Viewing Roles and Capabilities in SAS Management Console

To view details about a role, open the User Manager plug-in in SAS Management Console, right-click the role, and select **Properties**. You can then view tabs that display the role's members, capabilities, and contributing roles.

For example, the following display shows the capabilities for the Business Rules Manager: Rule Flow and Rule Set Designer role. These capabilities correspond to the description of this role in "Predefined Roles and Capabilities for SAS Decision Manager" on page 61.
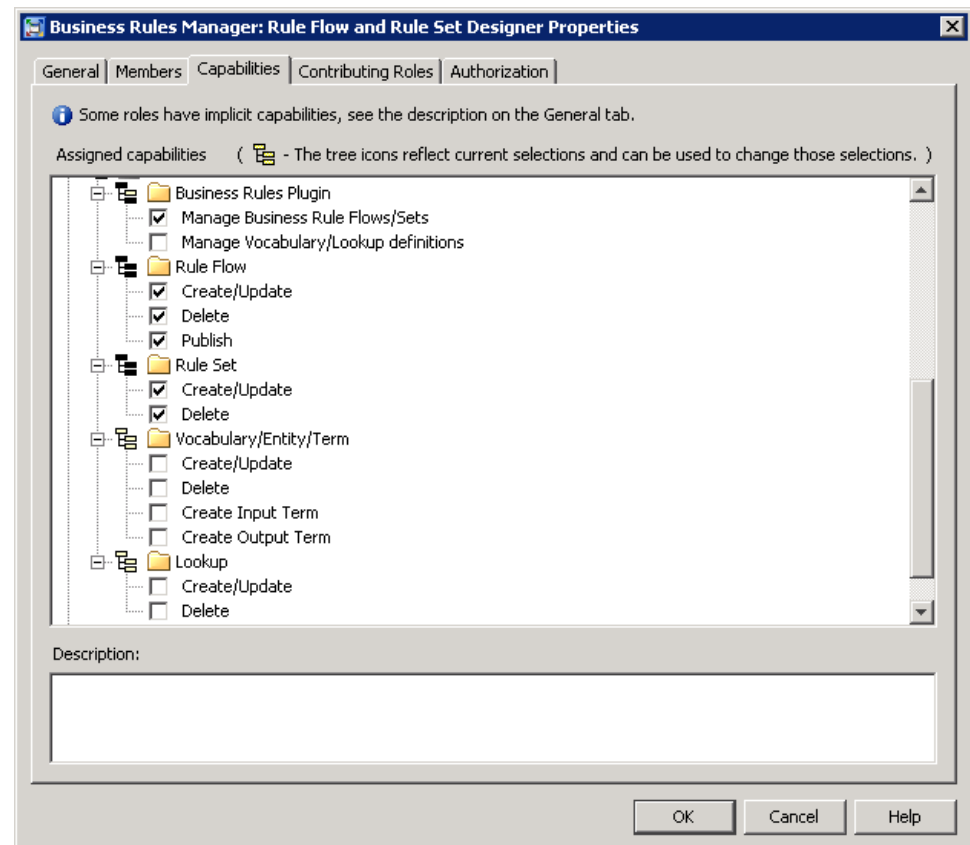
Note: Some roles have implicit capabilities that are not specified on the **Capabilities** tab.

The SAS Decision Manager capabilities control access to categories in the application. For example, the Rule Sets and Rule Flows categories do not appear when a user signs in to SAS Decision Manager if that user is not assigned to either of the following categories:

• Business Rules Manager: Rule Flow and Rule Set Designer

• Business Rules Manager: Rule Flow and Rule Set Read-Only

The Create/Update and Delete capabilities control access to specific object types. You can combine the category capabilities with the object capabilities to control access at whatever level is needed. For example, if you want a user to be able to view and edit rule flows only, the user should have only the following capabilities:

• Manage Business Rule Flows/Sets for the Business Rules Plugin

- Create/Update and Delete capabilities for Rule Flow objects

The following table describes the icons used in the Properties window.

Icon	Description
	None of the capabilities in this category have been specified for this role.
	Some of the capabilities in this category have been specified for this role, either explicitly or through a contributing role.
	All of the capabilities in this category have been specified for this role, either explicitly or through a contributing role.

Shaded check boxes indicate capabilities that come from contributing roles.

Adding a User to a Group or Role

In most cases, the best way to place a user in a role is to add the user to a group that belongs to the role. You can also add users directly to groups or roles.

To place a user in one of the predefined roles, you can add the user to one of the predefined groups. For example, to add a user to the Decision Manager Common: Administration role, add the user to the Decision Manager Common Administrators group.

For more information, see *SAS Management Console: Guide to Users and Permissions*.

Creating New Groups and Roles

The predefined groups and roles might be sufficient for many sites. Other sites might need to make application features available to users on either a broader or more granular basis than the predefined groups or roles allow.

You can use combinations of capabilities to create a new role. However, you can use only the capabilities that already appear in User Manager. You cannot create new capabilities.

For detailed information about roles and how to create them, see *SAS Management Console: Guide to Users and Permissions*.

Modifying Roles

The User Manager plug-in in SAS Management Console enables you to modify roles by selecting or deselecting different capabilities.

CAUTION:

No automated method can revert a role to its original set of capabilities. Instead of adjusting the capabilities of a predefined role, consider creating a new role. This advice is especially important if you need to make major changes.

If you modify a role, then follow these best practices:

- Do not rename the predefined roles. Renaming the predefined roles makes it difficult for SAS Technical Support to help you resolve problems.

- Keep a record of the changes that you make.

When modifying a role, you can use only the capabilities that already appear in User Manager. You cannot create new capabilities.

For more information about roles and how to modify them, see *SAS Management Console: Guide to Users and Permissions*.

Updating Groups and Roles for SAS Decision Manager 3.1

When you perform a migration or an upgrade, the groups, roles, and capabilities defined in SAS Management Console are preserved in case they have been customized for your site. To take advantage of new roles and capabilities available in SAS Decision Manager, you must modify the settings in SAS Management Console.

Create the Decision Builder User Role

When you install a new deployment of SAS Decision Manager 3.1 or migrate to SAS Decision Manager 3.1 from an earlier version, the Decision Builder User role is predefined for you. If you upgrade your existing deployment to SAS Decision Manager 3.1, you must create this role. For instructions about creating a new role in SAS Management Console, see "How to Create a Custom Role" in *SAS Management Console: Guide to Users and Permissions*.

* On the **Capabilities** tab, add the Decision and Decision Test capabilities to the role. These capabilities are under Decision Manager 3.1.
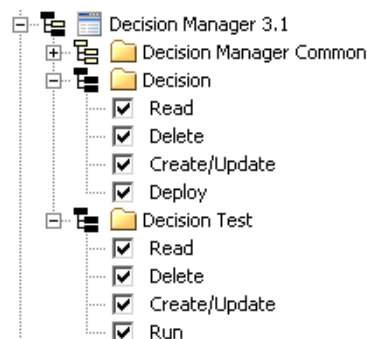
* On the **Members** tab, add the Decision Manager Users group to the role.

Update Capabilities for Business Rules Manager Roles

Beginning with SAS Decision Manager 2.2, new capabilities have been added. If you migrate or upgrade to SAS Decision Manager 3.1, you must manually add these new capabilities to take advantage of them. The new capabilities are Create Input Term, Create Output Term, and Publish.

The following table lists the roles to which these capabilities are assigned by default.

Role	Capabilities
Business Rules Manager: All Capabilities	Publish, Create Input Term, and Create Output Term

Role	Capabilities
Business Rules Manager: Rule Flow and Rule Set Designer	Publish
Business Rules Manager: Vocabulary and Lookup Designer	Create Input Term and Create Output Term

Model Management User Tasks

Overview of Model Management User Tasks

When you work in SAS Decision Manager, the application administrator assigns your user ID to one of three SAS Decision Manager groups: Model Manager Administrators, Model Manager Advanced Users, and Model Manager Users. Groups can perform certain tasks within SAS Decision Manager. For example, users in the Model Manager Administrator group are the only users who can lock a version.

Users in the Model Manager Administrator Users group can perform all task in the Models category view. The Model Manager Advanced Users and Model Manager Users groups are more restrictive. See the tables in the subsequent sections for a list of tasks and the groups whose users can perform the task.

An application administrator can create custom groups for your organization as well as assign roles to those groups. The pre–defined roles enable specific users or groups to be assigned in order to complete specific tasks within SAS Decision Manager. In most cases, roles are assigned to groups. Three of the roles are general and correspond to the groups that are supplied by SAS Decision Manager. Contact your application administrator to find out your group and roles.

The following table lists the abbreviations for groups that are used in the task tables below:

Group	Abbreviation
Model Manager Administrator Users	MM Admin
Model Manager Advanced Users	MM Adv User
Model Manager Users	MM User
Decision Manager Users	DCM User
Decision Manager Common Administrators	DCM Admin

Setting Up SAS Decision Manager

Use the following table to determine the users who can complete the tasks to set up SAS Decision Manager:

Task	Group
Create users in SAS Management Console	SAS Administrator
Create data libraries in SAS Management Console	DCM Admin, DCM User, MM Adv User, MM Admin, SAS Administrator
Create data libraries in the Data category view	DCM Admin, DCM User, MM Adv User, MM Admin, SAS Administrator
Create channel location folders on a SAS server	MM Admin
Create SAS channels in SAS Management Console	SAS Administrator
Define channel subscribers in SAS Management Console	SAS Administrator
Create project tables	MM User, MM Adv User, MM Admin
Register tables	DCM Admin, DCM User, MM Adv User, MM Admin
Configure the SAS Content Server for SAS Decision Manager	SAS Administrator
Create workflows	MM Admin, MM Adv User, MM User, DCM Admin, DCM User
Manage workflows	DCM Admin

Setting Up Projects and Portfolios

Use the following table to determine the users who can complete the tasks to set up projects and versions in SAS Decision Manager:

Task	Group
Create folders	MM Adv User, MM Admin
Create portfolios	MM Adv User, MM Admin
Create projects	MM Adv User, MM Admin
Create versions	MM Adv User, MM Admin
Delete folders, projects, and portfolios	MM Adv User, MM Admin
Archive and restore folders	MM Adv User, MM Admin.

Task	Group
Create and manage model and report templates	MM Adv User, MM Admin
	Note: An MM Adv User must have Write permissions to save a new template or changes to an existing template. For more information, see "Verify WebDAV Folder Permissions for User-Defined Templates" on page 17.
Create a workflow	MM User, MM Adv User, MM Admin
Assign participants to a workflow	DCM Admin
View workflows that are associated with a version	DCM Admin

Importing and Assessing Models

Use the following table to determine the users who can complete the tasks to import and assess models:

Task	Group
Import models	MM Adv User, MM Admin
Configure model properties	MM Adv User, MM Admin
View lineage and relationships for a model	MM User, MM Adv User, MM Admin
	Note: To view the relationships for a model in the SAS Lineage application, the Data Management: Lineage role must be associated with a user or group the user is in.
Map model variables to project variables	MM Adv User, MM Admin
Run model comparison and model validation reports	MM Adv User, MM Admin
Create user reports	MM Adv User, MM Admin
Create aggregated reports	MM Adv User, MM Admin
Create scoring output tables	MM Adv User, MM Admin
Create and run scoring tests	MM Adv User, MM Admin
Schedule a scoring test to execute	MM Adv User, MM Admin

Deploying and Delivering Models

Use the following table to determine the users who can complete the tasks to deploy and deliver models:

Task	Group
Set a champion model	MM Adv User, MM Admin
Flag a challenger model	MM Adv User, MM Admin
Validate the champion model by running a scoring test using test data and reviewing the scoring output	MM Adv User, MM Admin
Lock or unlock versions	MM Admin
Publish a project or model to a SAS channel	MM Adv User, MM Admin
Extract a champion model from a SAS channel	any user who has the appropriate access rights to the SAS Metadata Repository
Publish a model to the SAS Metadata Repository	MM Adv User, MM Admin
Publish a model scoring function or model scoring files to a database	MM Adv User, MM Admin

Monitor Champion Model Performance and Retrain Models

Use the following table to determine the users who can complete the tasks to create and run the reports that are used to monitor the champion model performance and to retrain models:

Task	Group
Set project properties	MM Adv User, MM Admin
Monitor performance of project champion models that are within a portfolio	MM Adv User, MM Admin
Edit a performance definition	MM Adv User, MM Admin
Schedule a performance definition to execute	MM Adv User, MM Admin
Execute the performance definition.	MM Adv User, MM Admin
Run performance monitoring batch jobs	in **Test** mode: MM User, MM Adv User, MM Admin in **Production** mode: MM Adv, MM Admin

Task	Group
View monitoring reports and charts	MM User, MM Adv User, MM Admin
Delete performance data sets	MM Adv User, MM Admin
Create and manage dashboard report definitions	MM Adv User, MM Admin
Generate dashboard reports	MM Adv User, MM Admin
View dashboard reports	MM User, MM Adv User, MM Admin
Edit a model retrain definition	MM Adv User, MM Admin
Execute or schedule a model retrain definition	MM Adv User, MM Admin
View retrained models and the associated model comparison reports	MM User, MM Adv User, MM Admin

General Tasks

Use the following table to determine the users who can complete these general tasks:

Task	Group or Role
Add attachments and comments	MM User, MM Adv User, MM Admin
Search for models	MM User, MM Adv User, MM Admin
Set the status of a project champion model and challenger models	MM Adv User, MM Admin
Replace a champion model	MM Adv User, MM Admin
View workflow tasks	MM User, MM Adv User, MM Admin A user must be the actual owner of a task or assigned the workflow participant role of potential owner or business administrator to view tasks in the My Tasks category
Work with workflow tasks	MM User, MM Adv User, MM Admin A user who is a workflow participant can claim, release, and complete tasks.

Chapter 7
Configuring SAS Workflow

Configuring SAS Workflow for Use with SAS Decision Manager

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 definitions. The SAS middle tier hosts the workflow engine and the workflow services as part of the SAS Web Infrastructure Platform. SAS Decision Manager is used to manage the workflows that are associated with versions. For more information about SAS Workflow, see "SAS Workflow" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Prerequisites for Using SAS Workflow

To use SAS Workflow with SAS Decision Manager, be sure the following prerequisites are 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. Users must be a member of the Decision Manager Common Administrators Group or another user group that is associated with the Decision Manager Common: Administration role. The users or groups must also be assigned to a workflow template management role. For more information, see "Deploying and Maintaining Workflows" in *SAS Workflow Studio: User's Guide*.

3. Workflow definitions must be created using SAS Workflow Studio. For more information about creating workflow definitions, see "Defining Workflows with SAS Workflow Studio" in *SAS Workflow Studio: User's Guide*.

Guidelines for Creating Workflow Definitions

When you create workflow definitions in SAS Workflow Studio to use with SAS Decision Manager, follow these guidelines:

• Participants, and policies must be added to the task level. Statuses added at the task level and the default statuses at the workflow definition level can be used for a task status. Data objects can be added at the workflow definition level or task level. Users can see only the data objects defined at the task level from the Workflows category in SAS Decision Manager.

• Only the Potential Owner and Business Administrator workflow roles are used by SAS Decision Manager and they can be used in either a participant or swimlane definition. The Actual Owner workflow role should not be used as part of a workflow definition.

• In order to assign additional participants to tasks in SAS Decision Manager, the user must have or be in a group that is assigned the workflow role of Business Administrator. Also, in order to manage workflows and assign participants, the user must be in the Decision Manager Common Administrators group, or in a group that is a member of the Decision Manager Common Administrators group or that is associated with the Decision Manager Common: Administration role in SAS Management Console.

The following groups are created at installation time:

  • Decision Manager Common Administrators Group

  • Decision Manager Users Group

• Only workflow definitions that are activated in the Workflow repository, that are associated with the `mmapi` tag attribute in the file properties, are available to SAS Decision Manager.

How to Add the Approval Attribute to a Status

To add the `Approval` attribute to a status:

1. Expand the **Statuses** folder in the **Workflow Tree**.

2. Right-click a status and select **Edit**.
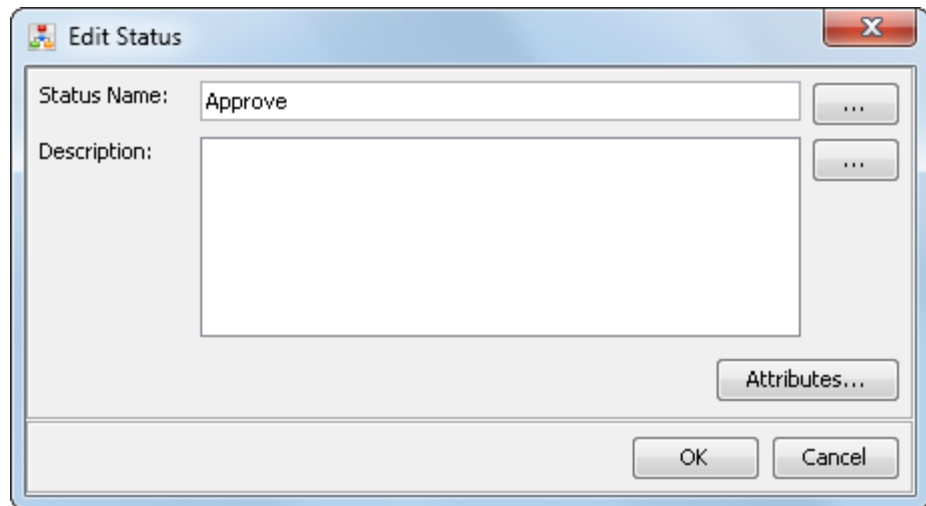
3. Click **Attributes**.

4. Click **Add** and enter the following values for the new attribute.

   Key
   > Approval

   > *Note:* This key is case-sensitive.

   Value
   > true

5. Click **OK** twice to save.

How to Make Workflow Definitions Available to SAS Decision Manager

After you have created a workflow definition in the SAS Workflow Studio, you must make the workflow definition available to SAS Decision Manager.

To save the workflow definition to the Workflow repository:

1. Save the workflow definition to your local drive.

2. Log on to the server.

3. Add the tag attribute of `mmapi` to the workflow definition file properties.

4. Upload the workflow definition.

5. Verify that the workflow definition is available in the Workflows category.

For more information, see "Deploying and Maintaining Workflows" in *SAS Workflow Studio: User's Guide*.

Log On to the Server

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

To log on to the server:

1. Select **Server ⇨ Log On**.

2. In the Log On window, select the host-name from the **SAS environment** drop-down list.

   Note: For more information, see "Configuring the SAS Environment File" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

3. Enter a user ID and password, and click **Log On**.

4. Click **OK** if a confirmation message appears.

Add Tag Attributes to a Workflow Definition

Only those workflow definitions in the Workflow repository that contain the `mmapi` tag attribute in the file properties are available to SAS Decision Manager.

To add a tag attribute to the file properties of a workflow template in SAS Workflow Studio:

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 Workflow Definition

To upload a workflow:

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 workflow definition.

3. Select the **Activate** option if you want to activate the current version in the Workflow repository.

4. Click **OK**.

5. Click **OK** if confirmation messages appear.

Verify That the Workflow Definitions Are Available In SAS Decision Manager

To verify that the workflow definitions are available in the Workflows category view of SAS Decision Manager:

1. Enter the URL http://*hostname*:*port*/SASDecisionManager in your web browser.

2. Enter the user ID and password for a user that is in the Decision Manager Common Administrators Group or a user group that is associated with the Decision Manager Common: Administration role.

3. Verify that the uploaded workflow definition is available in the Workflows category view. From the Workflows category view, select **Actions ⇨ Set Mappings**. The Set Mappings window appears with a list of the available workflow definitions.

For more information, see "Set Mappings" in *SAS Decision Manager: User's Guide*.

Configure Alert Notifications for SAS Workflow

To enable workflow participants to receive alert notifications from SAS Workflow, 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 tasks 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).

1. Log on to SAS Management Console as an 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, SAS Shared Services, and applications using SAS Workflow.

For more information about the notification properties, see "Setting Global Properties for SAS Applications" in *SAS Intelligence Platform: Middle-Tier Administration Guide*. For more information about setting the notification policies for SAS Workflow, see the SAS Workflow Studio Help or *SAS Workflow Studio: User's Guide*.

Chapter 8
Creating and Configuring Publication Channels

Overview of Creating and Configuring Publication Channels

SAS Decision Manager uses the SAS Publishing Framework to publish model updates to an operational environment for testing and production. The SAS Administrator creates and configures definitions for channels, content subscribers, and group subscribers. Then the user can use the SAS Decision Manager model extraction macros or user-written SAS code to retrieve and deploy the updated models to the operational environment.

As shown in the following figure, several tasks are necessary to configure and use the SAS Decision Manager publishing functionality.

Figure 8.1 *Configuring SAS Decision Manager to Enable Publishing Models*
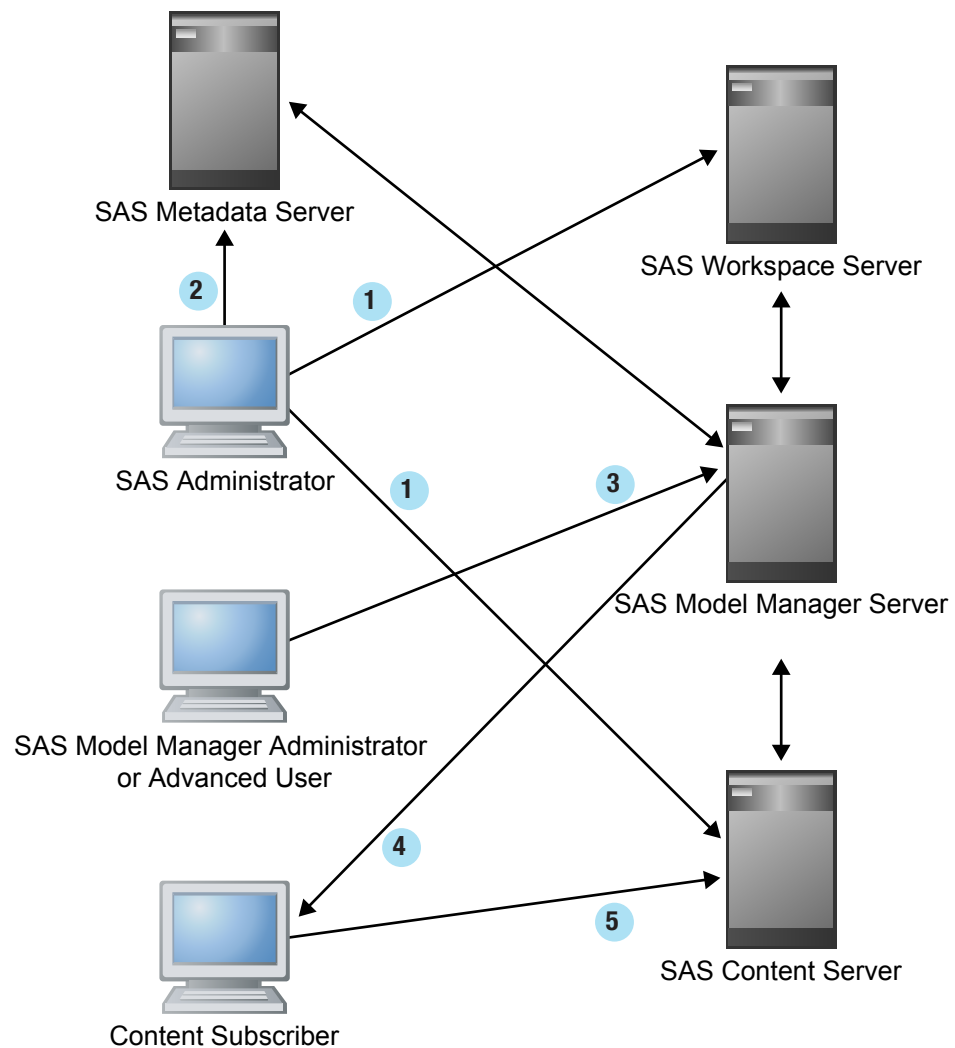
Here are the tasks.

1 The application administrator creates either an archive or a WebDAV persistent storage location for channels that is accessible from the SAS Workspace Server.

2 The application administrator creates users, HTTP servers, content subscribers, and channels using SAS Management Console.

3 The application administrator or an advanced user publishes models using SAS Decision Manager.

4 The content subscriber (for example, Scoring personnel) receives an e-mail notification from the server that contains a channel content update.

5 The content subscriber extracts models from a channel (for example, on a SAS Content Server) to prepare them for scoring.

Note: SAS Management Console Help provides details for your SAS Decision Manager publishing configuration options.

It is recommended that at first you use channels that have the Archive File type for the persistent storage option. This is the simplest channel definition and configuration to use to publish directly to your operational testing or production scoring servers. For example, during the installation of SAS Decision Manager, a channel called

MMChannel is created that has a persistent store Archive File type. For more information, see "Create a New Channel" on page 86.

Define an HTTP or HTTPS Server

The SAS Decision Manager installation process by default defines a SAS Web Server and a SAS Content Server. Use this process to add additional HTTP or HTTPS servers. A WebDAV-enabled HTTP or HTTPS content server must be defined in SAS Management Console before you can publish to channels from SAS Decision Manager. The server is usually a third-party server such as Microsoft Internet Information server or an Apache server.

Note: You must have WriteMetadata permission for a repository in order to define an HTTP or HTTPS content server for that repository.

To define your HTTP or HTTPS content server:

1. Start SAS Management Console. Open your existing connection profile for your server. If your connection profile is not available in the list, see SAS Management Console Help.

2. From the **Plug-ins** tab, right-click **Server Manager**, and then select **New Server**.

3. Select **Resource Templates** ⇨ **Servers** ⇨ **Content Servers** ⇨ **Http Server**, and then click **Next**.

   Note: If the HTTP server template is not available, then you must add the resource template. For more information, see the SAS Management Console Help.

4. Enter the name and the description of your HTTP server. Click **Next**.

5. (Optional) On the server properties page, enter the software version and vendor information for the third-party HTTP or HTTPS server that you are defining.

6. Click **New** to create a base path or paths on your server.

   Note: If you have not defined the base path for your HTTP server, see "Define Publish Locations for the SAS Content Server" on page 82.

7. In the **Base Path** field, specify the location of the top-level directory where report content items such as report definitions or image files are stored. (This path must be set up as an alias on the web server.) The **Description** field is optional.

8. Select the **Supports WebDAV** option and then click **OK** to save your settings. The new base path appears in the **Base Path(s)** field of the server properties page.

9. Click **Next**.

10. Enter the connection properties for your HTTP server:

    a. Select **DefaultAuth** from the list. When you click **New** to create a new domain, a dialog box appears. Enter the name and description of your domain.

    b. Enter the fully qualified name or the IP address of your server.

    c. Enter a port number (for example, 8080 for a web application server).

11. Click **Next**. The New Server Wizard window displays a summary of the settings for the new server and indicates that you have successfully completed the definition of a new server.

12. Click **Finish**. Your new server is displayed under the **Server Manager** node in the SAS Management Console Navigation Tree.

See Also

- SAS Management Console Help

- *SAS Intelligence Platform: Middle-Tier Administration Guide*

Define Publish Locations for the SAS Content Server

During the SAS Decision Manager installation process, the **ModelManager**, **sasfolders**, and **sasdav** WebDAV folders are automatically created on the SAS Content Server. You can use the SAS Content Server Administration Console (SCS Admin Console) to create a new publishing location for the WebDAV folder or to control access to an existing WebDAV folder. If you need to define a new WebDAV-enabled HTTP content server after the initial installation of SAS Decision Manager, then you must define a publishing location. For more information, see "Define an HTTP or HTTPS Server" on page 81.

Note: Although you can add a folder to the **sasfolders** location, the folder that you add is not added to the SAS Metadata Server.

T I P   The best practice is to add folders to metadata using SAS Management Console.

To define a new publishing location:

1. Access the SAS Content Server Administration console by entering the following URL in your web browser and substituting the server name and port number of your SAS Content Server: **http://*server-name:port*/SASContentServer/ dircontents.jsp**.

   Note: The default port number for the SAS Web Application Server is 80 for a UNIX environment and 7980 for a Windows environment.

2. Sign in to the console as an unrestricted user (for example, SAS Administrator). The SCS Admin Console window appears.

SCS Admin Console					sasadm@saspw Logout
Item name	**Primary type**	**Date created**	**Date modified**	**Delete**	**Permissions**
ModelManager	nt:davcollection	2015-05-21T15:19:02.308-04:00	2015-05-21T15:19:02.366-04:00	☐	▤
sascontent	nt:davcollection	2015-05-21T13:56:59.010-04:00	none	☐	▤
sasdav	nt:davcollection	2015-05-21T13:56:59.000-04:00	none	☐	▤
sasfolders	nt:davcollection	2015-05-21T13:56:59.010-04:00	none	☐	▤

Add folder

© SAS Institute Inc.
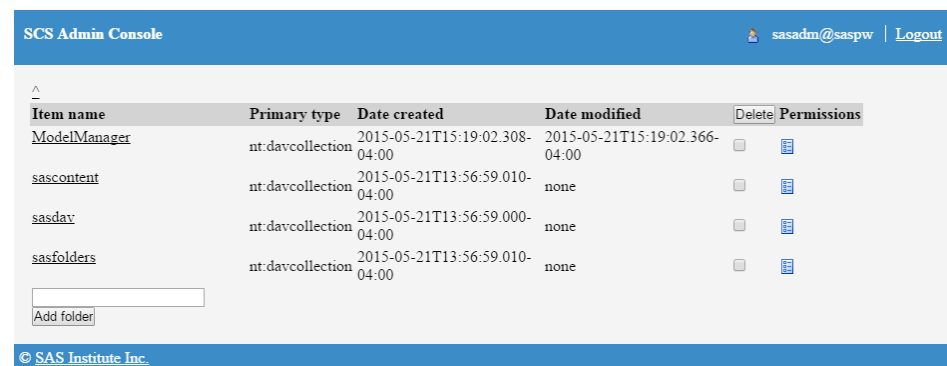
3. Enter a name for the folder in the text box and then click **Add folder** to create a new location for publishing channels.

4. (Optional) To create a subfolder, select the folder that you created in the previous step, enter a name for the subfolder in the text box, and click **Add folder**.

   Note: Use the breadcrumb trail above the list to return to a parent folder.

5. To set permissions for a folder:

a. Click the permission icon ▦ next to the item that you want to modify. The Permissions page appears.

b. For each principal that is listed, modify the permissions by changing each permission to **Yes** or **No**.

c. To add more principals to the page, do one of the following:

- If you know the principal's name, enter it in the field and click **Save changes**.

- Click **Search for Principals** to search for a name. When you find the principal that you want to add, select the check box that is next to the principal's name and then click **Return**.

    After the principal's name appears on the permission page, you can set permissions for the principal.

Note: For more information about administering the SAS Content Server, see *SAS Intelligence Platform: Web Application Administration Guide*.

Configuring Channels and Subscribers for SAS Decision Manager

Overview of Configuring Channels and Subscribers

The Publishing Framework plug-in to SAS Management Console enables you to administer the Publishing Framework.

Note: You need to verify that Publishing Framework plug-ins are available in your SAS Management Console navigation tree. If the plug-in is not available, you need to install SAS Foundation Services 1.3 or later so that you can configure your channels and subscribers for SAS Decision Manager.

With the Publishing Framework plug-in, you can manage subscribers and channels. For more information, see the Help.

When the Publishing Framework plug-in is available, the SAS Management Console Project Tree should look as follows:
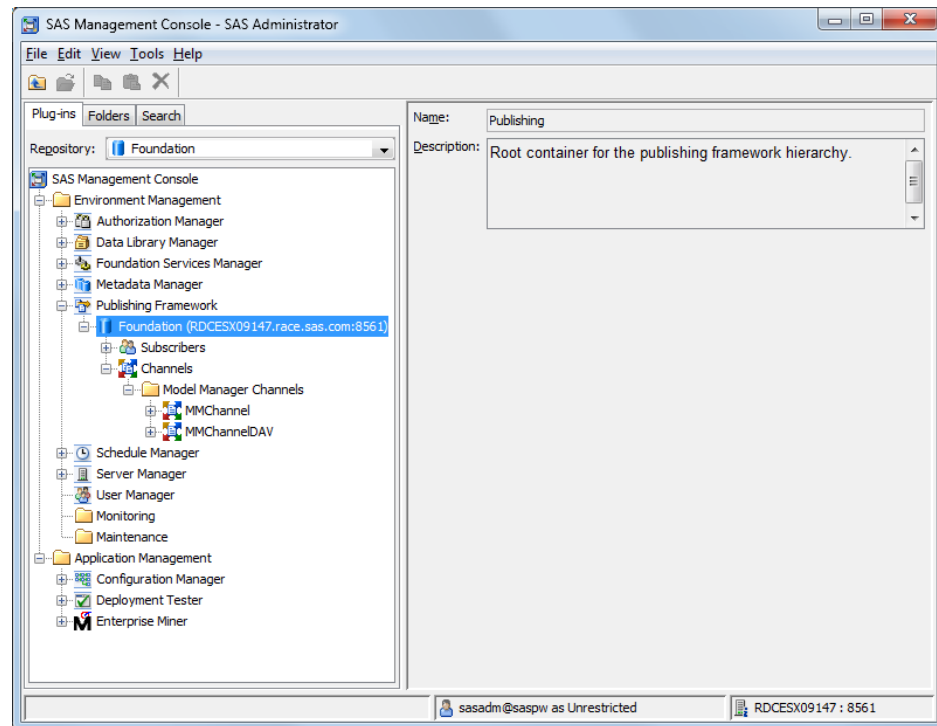
The SAS Metadata Server (for example, **Foundation**) that is shown under the Publishing Framework plug-in contains the **Subscribers** folder and the **Channels** folder.

The Publishing Framework plug-in to SAS Management Console provides wizards that enable you to create subscribers. When you create a subscriber with a wizard, the subscriber object that has the specified attributes is stored on the SAS Metadata Server.

For more information about Publishing Framework, see the *SAS 9.4 Publishing Framework: Developer's Guide*.

Channel to Subscriber Configuration

There are several ways to configure channels to publish your models to the channel subscribers.

Choose one of these options to define the method to use for publishing channels:

1. **None** - specifies to publish all content that is published to the channel directly to the subscribers (through e-mail). The content is not persisted.

2. **Archive** - specifies a path and an optional logical server for the location of the persistent storage. The **Archive File** option is recommended for publishing model packages. Publishing Framework publishes the content as an archive (binary) SPK (SAS package) file to the persistent storage location.

3. **WebDAV** - specifies the WebDAV server location.

TIP The best practice is to use the **Archive File** type for channel persistent storage and **e-mail** for subscriber notification.

Before publishing models using SAS Decision Manager, you must create channels and subscribers to publish your model updates.

Creating Channels and Subscribers

The channel sends the information from the publishers to the subscribers who want it.

A subscriber is a person or a program that has a need for information that is published. To receive information from a channel, the user must be defined as a subscriber.

The Publishing Framework plug-in provides wizards that enable you to create subscribers. Information about the subscriber is stored on the SAS Metadata Server.

Note: Channel subscribers must be users of the SAS Metadata Server and their e-mail addresses must be specified.

Create a Channel Folder

If you expect to create a large number of channels, then consider grouping related channels into channel folders. You can create subfolders within folders, thereby creating a folder hierarchy to which access controls can be applied. For more information, see the SAS Management Console Help.

Note: Currently it is not possible to move an existing channel into a folder or from one folder to another. Plan ahead to avoid deleting and re-creating channels.

To create channel folders:

1. From the SAS Management Console navigation tree, expand the **Publishing Framework** node.

2. Select and expand the desired metadata repository node.

3. If you are creating a top-level folder, then select **Channels**. If you are creating a subfolder, then navigate to and select the desired parent folder.
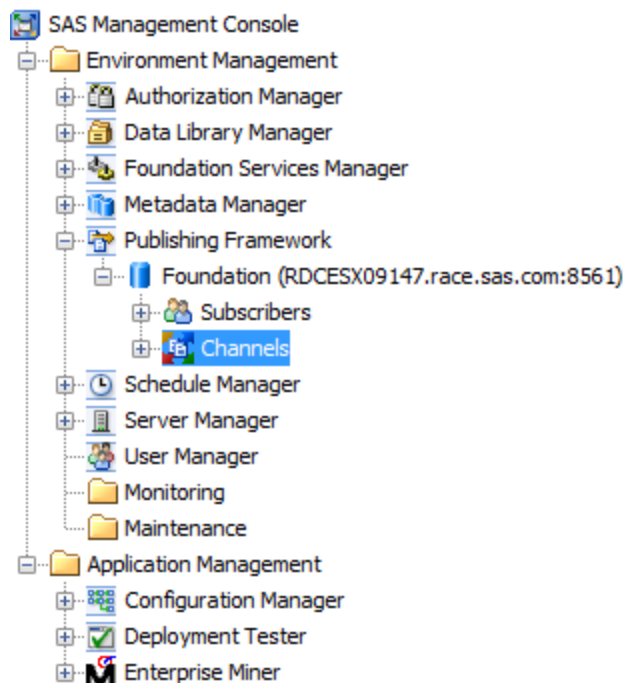
4. Right-click **Channels** and then select **New Folder**.

5. Enter a name for the new channel folder and then click **Next**. The new folder is created and the metadata definition information is displayed.

6. Click **Finish**.

Create a New Channel

To create a new channel:

1. From the SAS Management Console navigation tree, expand the **Publishing Framework** node.

2. Select and expand the desired metadata repository node.

3. If you are creating a channel within a folder, select the **Channels** node and navigate to the desired folder.

4. Right-click **Channels** or the desired channel folder and select **New Channel**.

5. Specify the name of your channel and click **Next**.

6. Use the arrow button to associate content subscribers with this channel to be notified at publish time. Click **Next**.

7. Select **Archive**.

8. Select **File** for **Archive Type** and enter the path of your publish location. Click **Next**.

   The information window appears, providing a summary of the input and status of successful completion of the channel creation.

   Note: Two other types, HTTP and FTP, are available for you to select from the list.

9. Click **Finish**. The new channel name is displayed under the **Channels** node of SAS Management Console.

For more information, see the SAS Management Console Help or *SAS 9.4 Publishing Framework: Developer's Guide*.

Create a New Subscriber

SAS Decision Manager supports only the content subscriber and the Name/Value pair filter for filtering. You can publish to a channel even when the channel does not have any associated subscribers. SAS Decision Manager users can extract contents from a channel if they are not subscribers of the channel. However, only subscribers of a channel can receive notifications. You can also create a subscriber group that contains individual subscribers or other subscriber groups. For more information, see the SAS Management Console Help.

To create a new content subscriber:

1. Expand the **Publishing Framework** node in the SAS Management Console navigation tree.

2. Select the desired metadata repository node.

3. Select **Subscribers ⇨ Content Subscribers**.

SAS Management Console
  Environment Management
    Authorization Manager
    Data Library Manager
    Foundation Services Manager
    Metadata Manager
    Publishing Framework
      Foundation (RDCESX09147.race.sas.com:8561)
        Subscribers
          Content Subscribers
        Channels
    Schedule Manager
    Server Manager
    User Manager
    Monitoring
    Maintenance
  Application Management
    Configuration Manager
    Deployment Tester
    Enterprise Miner

4. Right-click **Content Subscribers** and select **New Content Subscriber**.

5. Specify a name and a description for this subscriber. The name must be unique within its parent folder. The description is optional. Click **Next**.

6. Click **Select** to associate a person with this subscriber.

7. The search filter enables you to search the repository for users whose names either contain or are equal to a string that you specify. Enter the string in the text field, select either **contains** or **equals** from the list, and click **Search**. A list of users whose names meet your search criteria appears in the **Available People** list.

8. If the desired user does not exist in the repository, then click **New User** to define that user. Then, select the desired user from the **Available People** list and click **OK**.

9. Click **Next**.

10. Select the subscriber's delivery transport and then specify the attributes. Click **Next**.

11. Specify one or more filters to eliminate content that the subscriber does not want to receive. To add a filter, click the tab that corresponds to the type of filter (Name/ Value, Entry, or MIME Type). Select **Inclusion** or **Exclusion** and then click **Add** to specify the filter criteria.

12. Click **Next**.

13. Review the subscriber specifications. Click **Back** to make any corrections. Click **Finish** when you are satisfied with your selections.

For more information, see the Help or *SAS 9.4 Publishing Framework: Developer's Guide*.

Modify an Existing Channel or Channels Node Location

Modify the Directory Location for the Channels Node

To change the location of the application channels directory:

1. From SAS Management Console, expand the **Application Management** node on the **Plug-ins** tab.

2. Select and expand **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Enterprise Decision Manager 3.1**.

3. Right-click **Model Manager JavaSvcs 14.1** and select **Properties**.

4. Click the **Advanced** tab to modify the application channels directory. Change the property value for **App.ChannelDir** to a directory that is accessible by the SAS Workspace Server.

5. Click **OK**.

Modify the Persistent Store Directory Location for a Channel

To modify the location of the persistent store directory path for a channel:

1. From the SAS Management Console navigation tree, expand the **Publishing Framework** node.

2. Select and expand the desired metadata repository node.

3. If you are modifying a channel within a folder, select the **Channels** node and navigate to the desired folder.

4. Right-click the name of the channel that you want to modify, and then select **Properties**.

5. Click the **Persistent Store** tab, and modify the archive file path and server location.

6. Click **OK**.

See Also

SAS Management Console Help

Part 2

Macro Reference

Chapter 9

Macros for Importing and Exporting

Introduction to the Import and Export Macros

SAS Decision Manager provides the following macros for importing data into the SAS Decision Manager database and exporting data from the database. These macros must be run on the server tier.

%BRM_CREATE_TEMP_TERM
    reads a CSV file or a SAS data set and produces a SAS data set named WORK.TERM that can be used as input to the %BRM_LOAD_VOCABULARY macro.

%BRM_EXPORT_FOLDER
    exports definitions of business rules folders into a CSV file.

%BRM_EXPORT_LOOKUP
    exports the contents of lookup tables into a CSV file.

%BRM_EXPORT_RULE_FLOW
    exports rule flows from the SAS Decision Manager database into a CSV file.

%BRM_EXPORT_RULESET
    exports rule sets from the SAS Decision Manager database into a CSV file.

%BRM_EXPORT_VOCABULARY
    exports vocabularies from the SAS Decision Manager database into a CSV file.

%BRM_IMPORT_FOLDER
  imports folder definitions from a CSV file into the SAS Decision Manager database.

%BRM_IMPORT_LOOKUP
  imports lookup tables from a CSV file into the SAS Decision Manager database.

%BRM_IMPORT_RULE_FLOW
  imports rule flows from a CSV file into the SAS Decision Manager database.

%BRM_IMPORT_RULESET
  imports rule sets from a CSV file into the SAS Decision Manager database.

%BRM_IMPORT_VOCABULARY
  imports vocabulary terms from a CSV file into the SAS Decision Manager database.

%BRM_LOAD_VOCABULARY
  loads the vocabulary terms that are defined in the WORK.TERM data set into the SAS Decision Manager database.

Note: The same macro can be run simultaneously by multiple users. However, running import macros concurrently is not recommended.

Dictionary

%BRM_CREATE_TEMP_TERM

Reads a CSV file or a SAS data set that defines vocabulary terms and produces a SAS data set named WORK.TERM. You can use the WORK.TERM data set as input to the %BRM_LOAD_VOCABULARY macro.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_CREATE_TEMP_TERM (DATAFILE=%STR('*input_file*')<, BRM_USER=*user_ID*>);

Required Argument

DATAFILE=%STR('*input_file*')
  specifies either a SAS data set name or the full pathname to a CSV file. Enclose the filename in single quotation marks.

  If the input file is a CSV file, the first row of the file must contain valid SAS column names, and the remaining rows must contain column values. The column values can be numeric or character data only. You cannot specify SAS informats in the column data. The column names must be unique. For example, a simple CSV file that specifies two columns, both with numeric data, might look like the following:

```
patientID,BloodPressure
1,140
2,141
3,142
```

Optional Argument

BRM_USER=*user_ID*

specifies the user ID that you want to be associated with the data that is imported. This user ID is associated with the imported objects in the SAS Decision Manager database and is displayed in the interface.

Default   User ID of the user that is logged on to the server and running the macro

Details

This macro reads a CSV file or SAS data set that defines vocabulary terms and creates a SAS data set named WORK.TERM. You can use the WORK.TERM data set as input to the %BRM_LOAD_VOCABULARY macro. The %BRM_LOAD_VOCABULARY macro loads the vocabulary terms into the SAS Decision Manager database. See "%BRM_LOAD_VOCABULARY" on page 108 for more information.

The %BRM_CREATE_TEMP_TERM macro derives domain types and domain values for the vocabulary terms based on the data type of the term as described in Table 9.1.

Table 9.1   *Domain Types and Values for Input Terms*

Term Data Type	Derived Domain Type	Derived Domain Values
Character	Discrete	If there are ten or fewer distinct values in the input data, all of the values are included in the list of domain values. If there are greater than ten distinct values in the input data, individual values are not listed in the domain values.
Date	Continuous	No input values are included in the list of domain values.
Datetime	Continuous	No input values are included in the list of domain values.
Boolean	Boolean	True and False
Numeric	If there are ten or fewer distinct values in the input data, the domain type is Discrete. If there are greater than ten distinct values, the domain type is Continuous.	For Discrete domain types, all of the values in the input data are included in the list of domain values. For Continuous domain types, only the minimum and maximum values are included in the list of domain values.

%BRM_EXPORT_FOLDER

Exports definitions of business rules folders into a CSV file. You can modify the CSV file and use it as input to the %BRM_IMPORT_FOLDER macro.

Requirement: This macro must be run on the server tier.

Tip: You can modify the data values in the exported CSV file, and then re-import the data. However, do not modify the CSV file structure (column or row order).

Syntax

%BRM_EXPORT_FOLDER (CSV=%STR(*output_filename*.CSV)
<, FOLDER_PATH=<%STR(*pathname_1*<, *pathname_2>...*)>>);

Required Argument

CSV=%STR(*output_filename*)
    specifies the full pathname to the CSV file for the exported data.

Optional Argument

FOLDER_PATH= <%STR(*pathname_1* <, pathname_2> ...)>
    specifies the full pathname of the business rules folders that you want to export. Use a forward slash to separate folder names. By default, %BRM_EXPORT_FOLDER exports all business rules folders. You can explicitly specify that you want to export all folders by entering an empty folder list:

    FOLDER_PATH=

    You do not need to specify the FOLDER_PATH= option unless you want to export a specific folder.

    Example folder_path=%str(Applications, Retail/ApprovedLoans)

%BRM_EXPORT_LOOKUP

Exports the contents of lookup tables into a CSV file. You can modify the CSV file and use it as input to the %BRM_IMPORT_LOOKUP macro.

Requirement: This macro must be run on the server tier.

Tip: You can modify the data values in the exported CSV file, and then re-import the data. However, do not modify the CSV file structure (column or row order).

Syntax

%BRM_EXPORT_LOOKUP (CSV=%STR(*output_filename*.CSV)
<, FOLDER_PATH=%STR(*pathname*)>
<, LOOKUP=%STR('*lookup_table_1*'<, '*lookup_table_2*'>...)>);

Required Argument

CSV=%STR(*output_filename*)
    specifies the full pathname to the CSV file for the exported data.

Optional Arguments

FOLDER_PATH=%STR(*pathname*)
    specifies the full pathname to the business rules folder from which you want to export lookup tables. Use a forward slash to separate folder names.

If you specify a folder pathname, then only the lookup tables in that folder are exported. For example, if you specify FOLDER_PATH=%STR(Loans/Retail), then only the lookup tables in the folder Loans/Retail are exported. If you specify both LOOKUP=%STR('CountryCodes', 'ZipCodes') and FOLDER_PATH=%STR(Loans/Retail), but neither of the specified lookup tables are in the Loans/Retail folder, then no lookup tables are exported.

LOOKUP=%STR('*lookup_table_1*'<, '*lookup_table_2*'>...)
    specifies names of the lookup tables that you want to export. Enclose each table name in single quotation marks. Separate multiple names with commas.

    By default, %BRM_EXPORT_LOOKUP exports all lookup tables. You do not need to specify the LOOKUP= option unless you want to export specific tables.

    Example   lookup=%str('BadVINSTates','StateCodes')

%BRM_EXPORT_RULE_FLOW

Exports rule flows from the SAS Decision Manager database into a CSV file. You can modify the CSV file and use it as input to the %BRM_IMPORT_RULE_FLOW macro.

Requirement: This macro must be run on the server tier.

Tip: You can modify the data values in the exported CSV file, and then re-import the data. However, do not modify the CSV file structure (column or row order).

Syntax

%BRM_EXPORT_RULE_FLOW (
RULEFLOWS=ALL | %STR(*rule_flow_1*<,*rule_flow_2*>...),
CSV=%STR(*output_filename*.CSV)<, FOLDER_PATH=%STR(*pathname*)>);

Required Arguments

CSV=%STR(*output_filename*)
    specifies the full pathname to the CSV file for the exported data.

RULEFLOWS=ALL | %STR(*rule_flow_1*<, *rule_flow_2*>...)
    specifies the rule flows that you want to export. Specify ALL to export all rule flows. To export only selected rule flows, specify the identification numbers of the rule flows. Separate multiple identification numbers with commas.

    Example   ruleflows=%str(10168,10043)

Optional Argument

FOLDER_PATH=%STR(*pathname*)
    specifies a business rules folder from which you want to export rule flows. Use a forward slash to separate folder names.

    If you specify a folder pathname, then only the rule flows in that folder are exported. For example, if you specify RULEFLOWS=ALL and FOLDER_PATH= %STR(Loans/Retail), then only the rule flows in the folder Loans/Retail are exported. If you specify RULEFLOWS=%STR(10045,10572) and FOLDER_PATH=%STR(Loans/Retail), but neither of the specified rule flows are in the Loans/Retail folder, then no rule flows are exported.

%BRM_EXPORT_RULESET

Exports rule sets from the SAS Decision Manager database into a CSV file. You can modify the CSV file and use it as input to the %BRM_IMPORT_RULESET macro.

Requirement: This macro must be run on the server tier.

Tip: You can modify the data values in the exported CSV file, and then re-import the data. However, do not modify the CSV file structure (column or row order).

Syntax

%BRM_EXPORT_RULESET (RULESETS=ALL | %STR(*rule_set_1*<, *rule_set_2*>...), CSV=%STR(*output_filename*.CSV)<, FOLDER_PATH=%STR(*pathname*)>);

Required Arguments

CSV=%STR(*output_filename*)
specifies the full pathname to the CSV file for the exported data.

RULESETS=ALL | %STR(*rule_set_1*<, *rule_set_2*>...)
specifies the rule sets that you want to export. Specify ALL to export all rule sets. To export only selected rule sets, specify the identification numbers of the rule sets. Separate multiple identification numbers with commas.

Example rulesets=%str(10168,10043)

Optional Argument

FOLDER_PATH=%STR(*pathname*)
specifies the full pathname for the business rules folder from which you want to export rule sets. Use a forward slash to separate folder names.

If you specify a folder pathname, then only the rule sets in that folder are exported. For example, if you specify RULESETS=ALL and FOLDER_PATH=%STR(Loans/Retail), then only the rule sets in the folder Loans/Retail are exported. If you specify RULESETS=%STR(10045,10572) and FOLDER_PATH=%STR(Loans/Retail), but neither of the specified rule sets are in the Loans/Retail folder, then no rule sets are exported.

%BRM_EXPORT_VOCABULARY

Exports vocabularies from the SAS Decision Manager database into a CSV file. You can modify the CSV file and use it as input to the %BRM_IMPORT_VOCABULARY macro.

Requirement: This macro must be run on the server tier.

Tip: You can modify the data values in the exported CSV file, and then re-import the data. However, do not modify the CSV file structure (column or row order).

Syntax

%BRM_EXPORT_VOCABULARY (
VOCAB=ALL| %STR('*vocabulary_1*'<, '*vocabulary_2*'>...),
CSV=%STR(*output_filename*.CSV)<, FOLDER_PATH=%STR(*pathname*)>);

Required Arguments

CSV=%STR(*output_filename*)
    specifies the full pathname to the CSV file for the exported data.

VOCAB=ALL | %STR('*vocabulary_1*'<, '*vocabulary_2*'>...)
    specifies the names of the vocabularies that you want to export. Specify ALL to
    export all vocabularies. To export only selected vocabularies, specify the names of
    the vocabularies enclosed in quotation marks. Separate multiple names with
    commas.

    Example   vocab=%str('LRAutoVocab','AcmeAuto')

Optional Argument

FOLDER_PATH=%STR(*pathname*)
    specifies the full pathname of the business rules folder from which you want to
    export vocabularies. Use a forward slash to separate folder names.

    If you specify a folder pathname, then only the vocabularies in that folder are
    exported. For example, if you specify VOCAB=ALL and FOLDER_PATH=
    %STR(Loans/Retail), then only the vocabularies in the folder Retail are exported. If
    you specify VOCAB=%STR('loanVocab','riskVocabulary') and FOLDER_PATH=
    %STR(Loans/Retail), but neither of the specified vocabularies are in the Retail
    folder, then no vocabularies are exported.

%BRM_IMPORT_FOLDER

Imports the folder definitions in the specified CSV file into the SAS Decision Manager database.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_IMPORT_FOLDER (CSV=%STR(*input_filename*.CSV),
REJECT=%STR(*reject_filename*.CSV)<, BRM_USER=*user_ID*>);

Required Arguments

CSV=%STR(*input_filename*)
    specifies the full pathname to the CSV file where you want to import the data from.
    For more information, see "Format of the Folder CSV Input File" on page 98.

REJECT=%STR(*reject_filename*)
    specifies the full pathname to the CSV file where you want the macro to write any
    records that were not imported to the SAS Decision Manager database. See "Using
    the %BRM_IMPORT_FOLDER Macro" on page 98 for more information.

Optional Argument

BRM_USER=*user_ID*

> specifies the user ID that you want to be associated with the data that is imported. This user ID is associated with the imported objects in the SAS Decision Manager database and is displayed in the interface.

> **Default** User ID of the user that is logged on to the server and running the macro

Details

Using the %BRM_IMPORT_FOLDER Macro

The %BRM_IMPORT_FOLDER macro enables you to create new folders. You cannot update the content in existing folders with this macro. The macro uses the pathname to determine whether a folder already exists. If the pathname already exists, then the folder is rejected.

The %BRM_IMPORT_FOLDER macro runs several validation checks as it imports the folders. For example, it checks whether each folder pathname begins with a top-level folder and verifies that individual folder names are not longer than 100 characters. If the macro finds an invalid folder definition in the CSV file, it writes a message to the SAS log, and the folder is rejected. The macro writes the input records for the rejected folder to the CSV file that was specified in the REJECT= option.

Format of the Folder CSV Input File

Each row of the CSV input file identifies a folder. The CSV file must contain all of the columns listed in the following table, in the order listed. You must specify values for all columns, except as noted in the following table. To create a blank column in the CSV file, specify two comma separators without any content between them. For example, to create a folder named Applications and to specify a blank column for the folder description, specify the following in the CSV file:

```
Applications,,N,Loans/Retail
```

Table 9.2 *Format of the Folder CSV Input File*

Column	Description	Can Column Be Blank
FOLDER_NM	The name of the folder where you want to import the contents of the CSV file.	No
FOLDER_DESC	The description of the folder.	Yes
TOP_LEVEL_FOLDER_FLG	Specifies whether the folder is a top-level folder. Specify **Y** or **N**.	No
FOLDER_PATH	The pathname to the business rules folder where you want to import the contents of the CSV file. This pathname must exist. Separate folder names with forward slashes.	No

%BRM_IMPORT_LOOKUP

Imports lookup tables from the specified CSV file into the SAS Decision Manager database.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_IMPORT_LOOKUP (CSV=%STR(*input_filename*.CSV),
REJECT=%STR(*reject_filename*.CSV)<, BRM_USER=*user_ID*>);

Required Arguments

CSV=%STR(*input_filename*)
specifies the full pathname to the CSV file where you want to import the data from.
For more information, see "Format of the Lookup CSV Input File" on page 100.

REJECT=%STR(*reject_filename*)
specifies the full pathname to the CSV file where you want the macro to write any
records that were not imported to the SAS Decision Manager database. See "Using
the %BRM_IMPORT_LOOKUP Macro" on page 99 for more information.

Optional Argument

BRM_USER=*user_ID*
specifies the user ID that you want to be associated with the data that is imported.
This user ID is associated with the imported objects in the SAS Decision Manager
database and is displayed in the interface.

Default User ID of the user that is logged on to the server and running the macro

Details

Using the %BRM_IMPORT_LOOKUP Macro

The %BRM_IMPORT_LOOKUP macro enables you to do the following tasks:

• add new lookup tables

• add new key-value pairs to existing lookup tables

• update (refresh) existing key-value pairs in existing lookup tables

The macro uses the lookup table name and pathname to determine whether a lookup
table already exists. If the lookup table already exists, then it is updated. If the pathname
exists but the lookup table does not exist, the lookup table is created. If the pathname
does not exist, then the lookup table is rejected.

The %BRM_IMPORT_LOOKUP macro runs several validation checks as it imports the
lookup tables. For example, the macro checks whether the LOOKUP_NM or NAME
columns in the input file are empty or whether the LOOKUP_NM column specifies an
invalid lookup name. All valid key-value pairs are imported. If the macro finds an
invalid key-value pair in the CSV file, it writes a message to the SAS log, and the key-
value pair is rejected. The macro writes the input records for the rejected key-value pairs
to the CSV file that was specified in the REJECT= option.

Format of the Lookup CSV Input File

The first row of the input file must be a header row. SAS Decision Manager begins importing data with row two.

Each subsequent row of the CSV input file identifies a key-value pair and the lookup table in which it belongs. The CSV file must contain all of the columns listed in the following table, in the order listed. You must specify values for all columns, except as noted in the table. To create a blank column in the CSV file, specify two comma separators without any content between them. The following example specifies the keys **AU** and **CA** and associates them with the values **Australia** and **Canada**, respectively. These key-value pairs will be imported into the lookup table **Country_Codes**.

```
FOLDER_PATH,LOOKUP_NM,DESCRIPTION,NAME,VALUE
Loans/Retail,Country_Codes,,AU,Australia
Loans/Retail,Country_Codes,,CA,Canada
```

This input file would appear in Microsoft Excel as shown below.

	A	B	C	D	E
1	FOLDER_PATH	LOOKUP_NM	DESCRIPTION	NAME	VALUE
2	Loans/Retail	Country_Codes		AU	Australia
3	Loans/Retail	Country_Codes		CA	Canada

Note: When you import a lookup table with the %BRM_IMPORT_LOOKUP macro, the first line of the input file must be a header row.

Table 9.3   Format of the Lookup CSV Input File

Column	Description	Can Column Be Blank
FOLDER_PATH	The pathname to the business rules folder where you want to import the lookup table. This pathname must exist. Separate folder names with forward slashes.	No
LOOKUP_NM	The name of the lookup table.	No
DESCRIPTION	The description of the lookup table.	Yes
NAME	The lookup key.	No
VALUE	The lookup value.	Yes

%BRM_IMPORT_RULE_FLOW

Imports rule flows from the specified CSV file into the SAS Decision Manager database.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_IMPORT_RULE_FLOW (CSV=%STR(*input_filename*.CSV),
REJECT=%STR(*reject_filename*.CSV)<, BRM_USER=*user_ID*>)<, OVERWRITE=Y|
N>;

Required Arguments

CSV=%STR(*input_filename*
  specifies the full pathname to the CSV file where you want to import the data from.
  For more information, see "Format of the Rule Flow CSV Input File" on page 101.

REJECT=%STR(*reject_filename*)
  specifies the full pathname to the CSV file where you want the macro to write any
  records that were not imported to the SAS Decision Manager database. See "Using
  the %BRM_IMPORT_RULE_FLOW Macro" on page 101 for more information.

Optional Arguments

BRM_USER=*user_ID*
  specifies the user ID that you want to be associated with the data that is imported.
  This user ID is associated with the imported objects in the SAS Decision Manager
  database and is displayed in the interface.

  Default   User ID of the user that is logged on to the server and running the macro

OVERWRITE=Y|N
  specifies whether existing rule flows can be updated. If you specify **N**, the updates
  are rejected.

Details

Using the %BRM_IMPORT_RULE_FLOW Macro
The %BRM_IMPORT_RULE_FLOW macro enables you to add new rule flows and to
update existing rule flows. The macro uses the rule flow name and pathname to
determine whether a rule flow already exists. If the rule flow name and pathname
already exist, then the rule flow is updated (unless OVERWRITE=N is specified when
the macro is invoked). If the rule flow pathname exists but the rule flow name does not
exist, the rule flow is created. If the rule flow pathname does not exist, then the rule flow
is rejected.

The %BRM_IMPORT_RULE_FLOW macro runs several validation checks as it
imports the rule flows. For example, it checks whether a rule set is referenced in a given
rule flow more than once and whether section codes are correct. If the macro finds a
validation error in a rule flow, it writes a message to the SAS log, and the rule flow is
rejected. The macro writes the input records for the rejected rule flow to the CSV file
that was specified in the REJECT= option.

Format of the Rule Flow CSV Input File
Each row of the CSV input file identifies a rule set, and a rule flow provides the
information about how that rule set fits into the rule flow. The CSV file must contain all
of the columns that are listed in the following table, in the order listed. You must specify
values for all columns, except as noted in the table. To create a blank column in the CSV
file, specify two comma separators without any content between them.

For example, to add a rule set to position 1 in the main section of the rule flow named assignRisk in the **Retail/Loans** folder, you can specify the following in the CSV file:

```
.,assignRisk,,Y,main,Y,Loans/Retail,RuleSet1,Loans/Retail,Loan_Vocab,,1
```

Table 9.4 *Format of the Rule Flow CSV Input File*

Column	Description	Can Column Be Blank
RULE_FLOW_SK	The identification number of the rule flow.	Yes
RULE_FLOW_NM	The name of the rule flow where you want to add the rule set that is specified in RULE_SET_NM.	No
RULE_FLOW_SHORT_DESC	The description of the rule flow.	Yes
RULE_FIRED_OUTPUT_FLG	Specifies whether to create output only for records that fire rules. Specify **Y** or **N**. For some types of applications, only the output records for which at least one rule has fired are of interest. Limiting output is useful for applications that detect outliers, such as applications that detect fraud.	No
RULE_SET_SECTION_CODE	The section of the rule flow to which the rule set that is specified in RULE_SET_NM belongs. Specify **init**, **groupstart**, **main**, **groupend**, or **final**. The codes **groupstart** and **groupend** are valid only if you also specify at least one term for BY_TERM. See "Simple Rule Flows, Complex Rule Flows, and BY Groups" in *SAS Decision Manager: User's Guide* for more information.	No
INCLUDE_NODE_ OBJECT_FLG	Specifies whether the rule set specified in the RULE_SET_NM field is run when the rule flow executes. Specify **Y** or **N**. Selectively running certain rule sets is useful during rule flow development and testing.	No
RULE_FLOW_PATH	The pathname to the business rules folder for the rule flow. This pathname must exist. Separate folder names with forward slashes.	No
RULE_SET_NM	The name of the rule set to be added to the rule flow. A rule set can be added to the same rule flow only once.	No
RULE_SET_PATH	The pathname to the business rules folder for the rule set that is specified by RULE_SET_NM. The rule set must exist at the specified location. Separate folder names with forward slashes.	No
VOCAB_NM	The name of the vocabulary that the rule set uses. All rule sets in the same rule flow must use the same vocabulary.	No

Column	Description	Can Column Be Blank
BY_TERM	The list of BY-group terms that the rule set uses. Separate multiple BY-group terms with commas.	Yes
	The BY-group terms must be the same for all rule sets that are in the same rule flow. All of the BY-group terms must belong to the same vocabulary.	
	See "Simple Rule Flows, Complex Rule Flows, and BY Groups" in *SAS Decision Manager: User's Guide* for more information.	
ORDER	The order number for the rule set that is in the rule flow. Order numbers must start with 1 and be continuous through the entire rule flow. Do not restart order numbers at section boundaries.	No

%BRM_IMPORT_RULESET

Imports rule sets from the specified CSV file into the SAS Decision Manager database.

Requirements: The vocabulary used by the rule set must exist.

This macro must be run on the server tier.

CAUTION: **If the server is running in lockdown mode, you should not edit input files that are generated by the %BRM_EXPORT_RULESET macro before you import the data.** If the input file contains errors, the database might become corrupted. See "Locked-Down Servers" in *SAS Intelligence Platform: Security Administration Guide* and "LOCKDOWN Statement" in *SAS Intelligence Platform: Application Server Administration Guide* for more information.

Syntax

%BRM_IMPORT_RULESET (CSV=%STR(*input_filename*.CSV), REJECT=%STR(*reject_filename*.CSV)<, BRM_USER=*user_ID*> <, OVERWRITE=Y|N><, LOCK=Y|N>);

Required Arguments

CSV=%STR(*input_filename*)
   specifies the full pathname to the CSV file where you want to import the data from. For more information, see "Format of Rule Set CSV Input File" on page 104.

REJECT=%STR(*reject_filename*)
   specifies the full pathname to the CSV file where you want the macro to write any records that were not imported to the SAS Decision Manager database. See "Using the %BRM_IMPORT_RULESET Macro" on page 104 for more information.

Optional Arguments

BRM_USER=*user_ID*
   specifies the user ID that you want to be associated with the data that is imported. This user ID is associated with the imported objects in the SAS Decision Manager database and is displayed in the interface.

Default	User ID of the user that is logged on to the server and running the macro

LOCK=Y|N

specifies whether to lock the imported rule set.

Default	N

OVERWRITE=Y|N

specifies whether existing rule sets can be updated. If you specify **N**, the updates are rejected.

Default	N

Details

Using the %BRM_IMPORT_RULESET Macro

The %BRM_IMPORT_RULESET macro enables you to add new rule sets and to update existing rule sets. The macro uses the rule set name and rule set pathname to determine whether a rule set already exists. If the rule set pathname and name already exist, then the rule set is updated. If the rule set pathname exists but the rule set name does not exist, the rule set is created. If the rule set pathname does not exist, then the rule set is rejected.

The %BRM_IMPORT_RULESET macro runs several validation checks as it imports the rule sets. For example, it verifies that the expressions are valid, ensures that the first rule in each rule set uses the IF operator, and verifies that the specified vocabularies exist. If the macro finds a validation error in a rule set, it writes a message to the SAS log, and the rule set is rejected. The macro writes the input records for the rejected rule set and the reason that the record was rejected to the CSV file that was specified in the REJECT= option.

Rule sets that you import with the %BRM_IMPORT_RULESET macro are imported as unlocked versions. Before you can publish rule flows that contain the imported rule sets, you must lock the rule sets.

Format of Rule Set CSV Input File

Each row of the CSV input file specifies a rule, rule set, term, and an expression for that term. The row also specifies whether the expression is a condition expression or an action expression. Each row of the input file can specify only one condition expression or one action expression for a given rule. The CSV file must contain all of the columns that are listed in the following table, in the order listed. You must specify values for all columns, except as noted in the table. To create a blank column in the CSV file, specify two comma separators without any content between them.

For example, the following two lines add a rule to the rule set named riskSet, which uses the Loan_Vocab vocabulary. The first line adds the condition term **CondTerm** and assigns to it the expression **<5000**. The second line adds the action term **ActionTerm** and assigns to it the expression **'Bad'**.

```
.,riskSet,,Loan_Vocab,Loans/Retail,RuleName1,,1,if,Y,CondTerm,<5000,1,CONDITION
.,riskSet,,Loan_Vocab,Loans/Retail,RuleName1,,1,if,Y,ActionTerm,'Bad',1,ACTION
```

Table 9.5 *Format of the Rule Set CSV Input File*

Column	Description	Can Column Be Blank
RULE_SET_SK	The identification number of the rule set.	Yes
RULE_SET_NM	The name of the rule set where you want to add the rule that is specified in RULE_NM.	No
RULE_SET_DESC	The description of the rule set.	Yes
VOCAB_NM	The name of the vocabulary that the rule set uses. All rules in the same rule set must use the same vocabulary.	No
RULE_SET_PATH	The pathname to the business rules folder for the rule set. This pathname must exist. Separate folder names with forward slashes.	No
RULE_NM	The name of the rule to be added to the rule set.	No
RULE_DESC	The description of the rule.	Yes
RULE_SEQ_NO	The order number for the rule that is in the rule set. Order numbers in a rule set start with 1.	No
CONDITIONAL_NM	The operator for the rule. Specify **if**, **elseif**, or **or**. The first rule in a rule set must use the **if** operator. For information about these operators, see "Controlling Which Conditions Are Evaluated" in *SAS Decision Manager: User's Guide*.	No
RECORD_RULE_ FIRED_FLG	Specifies whether a rule-fired record is created when the condition for the rule identified in the RULE_NM field evaluates to **True**. Specify **Y** or **N**. If you specify **N**, a rule-fired record is not created regardless of what the condition evaluates to.	No
LHS_TERM	The term for the expression specified in the EXPRESSION column. Terms that are specified in the LHS_TERM column are the condition and action terms for the rule set. In the rule set editor, these terms appear in the Term column in the list view. They appear as column headings in the horizontal view and as row headings in the vertical view.	No

Column	Description	Can Column Be Blank
EXPRESSION	A single condition or action expression for the term specified in the LHS_TERM column. This expression is the expression that you would enter into a cell in the rule set editor. Enclose character strings in quotation marks. To specify a missing expression, enter a period and an underscore (._).See "Defining New Rules in the Rule Set" in *SAS Decision Manager: User's Guide* for more information about expressions.	Yes
EXPRESSION_ORDER	The order number of the rule's condition or action expressions. A rule's condition and action expressions are numbered beginning with 1. For example, a rule might have two condition expressions that are numbered 1 and 2, and it might have three action expressions that are numbered 1, 2, and 3.	No
EXPRESSION_TYPE	The type of expression. Specify **condition** or **action**.	No

%BRM_IMPORT_VOCABULARY

Imports vocabulary terms from the specified CSV file into the SAS Decision Manager database.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_IMPORT_VOCABULARY (CSV=%STR(*input_filename*.CSV), REJECT=%STR(*reject_filename*.CSV)<, BRM_USER=*user_ID*>);

Required Arguments

CSV=%STR(*input_filename*)
    specifies the full pathname to the CSV file that defines the vocabulary that you want to import. For more information, see "Format of the Vocabulary CSV Input File" on page 107.

REJECT=%STR(*reject_filename*)
    specifies the full pathname to the CSV file where you want the macro to write any records that were not imported to the SAS Decision Manager database. See "Using the %BRM_IMPORT_VOCABULARY Macro" on page 107 for more information.

Optional Argument

BRM_USER=*user_ID*
    specifies the user ID that you want to be associated with the data that is imported. This user ID is associated with the imported objects in the SAS Decision Manager database and is displayed in the interface.

Default	User ID of the user that is logged on to the server and running the macro	

Details

Using the %BRM_IMPORT_VOCABULARY Macro

The %BRM_IMPORT_VOCABULARY macro enables you to add new vocabulary terms. You cannot use the macro to update existing terms.

The %BRM_IMPORT_VOCABULARY macro runs several validation checks as it imports the vocabulary terms. For example, it verifies that term, entity, and vocabulary names are valid, and ensures that a term is not duplicated in a vocabulary. If the macro finds a validation error, it writes a message to the SAS log, and the term is rejected. The macro writes the input records for the rejected term to the CSV file that was specified in the REJECT= option.

Format of the Vocabulary CSV Input File

Each row of the CSV input file defines a term, including the term data type, domain type, and the entity and vocabulary that contains the term. The CSV file must contain all of the columns listed in the following table, in the order listed. You must specify values for all columns, except as noted in the table. To create a blank column in the CSV file, specify two comma separators without any content between them.

For example, the following lines add two terms to the Loan_Vocab vocabulary. The first term is named Priority, and it is an integer with domain values in the range 1–10. The second term is named RiskCategory, and it is a character string with domain values **'Low'** and **'High'**.

```
Loan_Vocab,,ApplicationEnt,,Priority,,Integer,discrete,(1-10),N,N,Loans/Retail
Loan_Vocab,,ApplicationEnt,,RiskCategory,,Character,discrete,(Low;High),N,N,Loans/Retail
```

Table 9.6 *Format of the Vocabulary CSV Input File*

Column	Description	Can Column Be Blank
VOCAB_NM	The name of the vocabulary where you want to add entity and term specified by VOCAB_ENTITY_NM and VOCAB_TERM_NM.	No
VOCAB_SHORT_ DESC	The description of the vocabulary.	Yes
VOCAB_ENTITY_NM	The name of the entity that the term in the VOCAB_TERM_NM column belongs to.	No
VOCAB_ENTITY_ SHORT_DESC	The description of the entity.	Yes
VOCAB_TERM_NM	The name of the term.	No
VOCAB_TERM_ SHORT_DESC	The description of the term.	Yes
VOCAB_TERM_DATA _TYPE_TXT	The data type of the term. Specify **Character**, **Decimal**, **Integer**, **Boolean**, **Date**, or **Datetime**.	No

Column	Description	Can Column Be Blank
VOCAB_TERM_ DOMAIN_TYPE_TXT	The domain type for the term. Specify **discrete**, **continuous**, or **Boolean**. A domain value is discrete if it is just an individual value such as **5.3** or **18JUL2012:10:25:00**. A domain value is continuous if it specifies a range such as **>5** or **<18JUL2012:10:25:00**. Terms that are assigned the data type **Character** can have discrete domain values only. Boolean terms can have Boolean domain values only.	No
VOCAB_TERM_ DOMAIN_TXT	The set of expected values for a term. Separate individual domain values with a semi-colon (;). See "Specify Domain Values" in *SAS Decision Manager: User's Guide* for more information.	Yes
VOCAB_TERM_ INPUT_EXCLUDE_ FLG	Specifies whether the term must be mapped to a column in an input data set. Specify **Y** or **N**.	No
VOCAB_TERM_ OUTPUT_EXCLUDE_ FLG	Specifies whether to exclude the term from the output data sets created by rule flows. Specify **Y** or **N**.	No
FOLDER_PATH	The pathname to the business rules folder for the rule flow. This pathname must exist. Separate folder names with forward slashes.	No

%BRM_LOAD_VOCABULARY

Loads the vocabulary terms that are defined in the WORK.TERM data set into the SAS Decision Manager database. You can create the WORK.TERM data set by using the %BRM_CREATE_TEMP_TERM macro.

Requirement: This macro must be run on the server tier.

Syntax

%BRM_LOAD_VOCABULARY (FOLDER_PATH=%STR(*pathname*),
VOCAB_NM=%STR(*vocabulary_name*),
VOCAB_ENTITY_NM=%STR(*entity_name*)<, BRM_USER=*user_ID*>);

Required Arguments

FOLDER_PATH=%STR(*pathname*)
   specifies the pathname to the business rules folder where you want to import the vocabulary terms. Separate folder names with forward slashes.

   Requirement   The pathname must exist. If the pathname does not exist, the macro terminates and writes an error message to the SAS log.

   Example   folder_path=%str(Loans/Retail/Applications)

VOCAB_NM=%STR(*vocabulary_name*)
    specifies the name of the vocabulary to which the terms in the WORK.TERM file will be added.

    Requirement   The vocabulary must not exist. If it already exists, the macro terminates and writes an error message to the SAS log.

VOCAB_ENTITY_NM=%STR(*entity_name*)
    specifies the name of the entity to which the terms in the WORK.TERM file will be added.

    Requirement   This entity must not exist. If it already exists, the macro terminates and writes an error message to the SAS log.

Optional Argument

BRM_USER=*user_ID*
    specifies the user ID that you want to be associated with the data that is imported. This user ID is associated with the imported objects in the SAS Decision Manager database and is displayed in the interface.

    Default   User ID of the user that is logged on to the server and running the macro

Chapter 10
Macros for Running Rule Flows

Introduction to Macros for Running Rule Flows

SAS Decision Manager provides the following macros for running rule flows. These macros must be run on the server tier.

%BRM_GET_RULE_FLOW_CODE
    creates a package that contains the SAS code for a rule flow.

%BRM_RULE_FLOW
    runs a rule flow.

Dictionary

%BRM_GET_RULE_FLOW_CODE

Creates a package that contains the SAS code for a specific rule flow. You can run this rule flow package by using the %BRM_RULE_FLOW macro.

    Requirement: This macro must be run on the server tier.

Syntax

%BRM_GET_RULE_FLOW_CODE (RULEFLOW_NAME=%STR(*name*), RULEFLOW_SK=*number*, FOLDER_PATH=%STR(*pathname*), FILELOCATION=%STR(*package_pathname*)<, WEBAUTHDOMAIN= %STR(*domain*)>);

Required Arguments

RULEFLOW_NAME=%STR(*name*)

specifies the name of the rule flow that you want to export.

Example ruleflow_name=%str(Ruleflow1)

RULEFLOW_SK=*number*

specifies the identification number of the rule flow. The identification number is shown in parentheses after the rule flow name on the rule flow **History** page or in the **Properties** section of the **Results** tab on the rule flow **Tests** page.

Example ruleflow_sk=10014

FOLDER_PATH=%STR(*pathname*)

specifies the full pathname to the business rules folder where the rule flow is defined. Separate folder names with forward slashes.

Example folder_path=%str(Claims/Processing)

FILELOCATION=%STR(*package_pathname*)

specifies the full pathname to the file for the DS2 package that is produced by the macro. The pathname must exist.

Example filelocation=%str(C:\MgrApprovals\approvalFlow.sas)

Optional Argument

WEBAUTHDOMAIN=%STR(*domain*)

specifies the authentication domain for the user ID that is used to access the rule flow.

Default DefaultAuth

Requirement A password must be defined for the user ID.

%BRM_RULE_FLOW

Runs rule flows. You can use the %BRM_RULE_FLOW macro to run packages that were created with the %BRM_GET_RULE_FLOW_CODE macro.

Requirement: This macro must be run on the server tier.

Syntax

FILENAME *fileref* "*pathname*";
%BRM_RULE_FLOW (INPUTTABLE=*libref.table_name*,
MAPPING=*mapfile*.mapping,
FILELOCATION=*fileref*, RULEFIRE=Y | N<, THREADCOUNT=*number*>);

Required Arguments

INPUTTABLE=*libref.table_name*
  specifies the libref and table name for the input table against which you want to run the rule flow.

MAPPING=*mapfile.mapping*
  specifies the file that contains the variable mappings. This file is typically a SAS file. See "Creating a Mapping Table" on page 114.

FILELOCATION=*fileref*
  specifies the fileref for the file that contains the DS2 package code for the rule flow. See *SAS Statements: Reference* for information about the FILENAME statement and how to define filerefs.

RULEFIRE=Y | N | S | D | Q
  specifies whether rule-fired data is recorded when the rule flow is run.

  Y   records both summary and detailed rule-fired data.

  N   does not record any rule-fired data.

  S   records only summary rule-fired data.

  D   records only detailed rule-fired data.

  Q   collects rule-fired data but does not generate summary or detailed rule-fired tables. The rule-fired data is added to the output table in columns named Rule Fired Count and _RULEFIREDCOUNTS_1.

Optional Arguments

CODETYPE=DS1|DS2
  determines whether SAS Decision Manager generates DS2 code or DATA step (DS1) code for rule flows. In many cases, you will get better performance by specifying DS1. However, consider specifying DS2 if your input data is in Teradata, Greenplum, or Hadoop, and you have installed the SAS Code Accelerator.

  Default DS2

THREADCOUNT=*number*
  specifies the number of processors that are available for concurrent processing. If the rule flow contains rules in either the INIT or FINAL sections, the value of the THREADCOUNT option is set to 1 when rule flow tests are run. This option is ignored when rule flows are executed in the database.

  Default the value of the CPUCOUNT= system option

  See "CPUCOUNT= System Option" in *SAS System Options: Reference*

Details

Running Rule Flows Dynamically
You can use the &DCM_USE_LATEST_VERSION and either the &DCM_RULEFLOW_NAME or &DCM_DEPLOYED_RULEFLOW_NAME macro variable to ensure that when a rule flow is run, the latest version of the rule flow is always used. If you specify both &DCM_RULEFLOW_NAME and &DCM_DEPLOYED_RULEFLOW_NAME, then the name specified by &DCM_DEPLOYED_RULEFLOW_NAME is used.

For &DCM_DEPLOYED_RULEFLOW_NAME, specify the name of the published rule flow and the identification number of the rule flow. You can find the published name and identification number in the **Name** column of the rule flow **History** page. For example:

```
%let DCM_DEPLOYED_RULEFLOW_NAME=published_flow_name(ID_number);
```

Note: If you specify &DCM_RULEFLOW_NAME and SAS Decision Manager finds multiple rule flows that match the specified name, it writes an error message in the SAS log, and the rule flow is not executed. If you encounter this issue, specify the specific rule flow by using &DCM_DEPLOYED_RULEFLOW_NAME.

Define these macro variables in preprocessing code such as the **Preprocessing Code** section of a rule flow test or in the **Precode** section of the **Precode and Postcode** tab in SAS Data Integration Studio. Define these variables before calling the %BRM_RULE_FLOW macro. For example:

```
%let DCM_USE_LATEST_VERSION=Y;
%let DCM_RULEFLOW_NAME=rule_flow_name;
```

Note: SAS Data Integration Studio uses the latest version of the rule flow that matches the variable mappings in the Business Rules transformation. SAS Decision Manager writes a note in the SAS log that states which version was selected.

Creating a Mapping Table

Note: You must create a mapping table only if you are invoking the %BRM_RULE_FLOW macro in SAS code. In SAS Data Integration Studio and in the SAS Decision Manager test interface, the mapping table is created for you.

You must supply a file that maps terms in the rule flow to columns in the input table. You can create this file manually, or you can create and run a rule flow test in SAS Decision Manager. The mapping tables that are created when a rule flow test is run are written to the WORK library. The code that produces the mapping table is written to the SAS log.

The mapping table also defines the names and structure of the output table, the rule-fired summary table, the rule-fired details table, and the test information table that are generated by the rule flow. The structure of the rule-fired summary table, rule-fired details table, and test information table is static, and you must define them as shown in "Example: Creating a Mapping File for a Simple Rule Flow" on page 115..

The number in the data set ID column in the example specifies which table the column that is being defined belongs to. The following table lists the possible values for this column and the default table names that are generated when a rule flow is run in SAS Data Integration Studio.

Data Set ID	Table	Contents	Name Generated by SAS Data Integration Studio
1	Rule-fired details	One row for each time that a rule evaluates to true. There might be multiple entries for the same rule, but each entry has different values for the _recordCorrelationKey and RULE_ACTION_FIRE_ID columns.	DCM_RULE_ACTION_FIRE
2	Test information	A single record that holds aggregate information about the execution of the rule flow.	DCM_DEPLOYMENT_EXECUTION
4	Input	Input data	

Data Set ID	Table	Contents	Name Generated by SAS Data Integration Studio
5	Output	Output data	
6	Rule-fired summary	A summary of how many times each rule fired.	DCM_RULE_FIRE_SUMMARY

Note: See "Rule-Fired Table Fields" in *SAS Decision Manager: User's Guide* for information about the columns in these tables.

Example: Creating a Mapping File for a Simple Rule Flow

The following example creates a mapping table that maps terms in the rule flow to an input table with five columns. The column names are EngineSize, Make, Model, MSRP, and Type.

The example assumes that the following librefs have been defined: RULEFIRE, DEPLOY, INDATA, and OUTLIB. It uses the table names listed the following table.

Data Set ID	Table	Libref and Table Name
1	Rule-fired details	RULEFIRE.Details
2	Test information	DEPLOY.ThisRun
4	Input	INDATA.InData
5	Output	OUTLIB.MyResults
6	Rule-fired summary	RULEFIRE.Summary

```
data work.MAPPING;
    attrib table length = $100;
    attrib column length = $100;
    attrib termid length = $100;
    attrib type length = $100;
    attrib datasetid length = $100;
    attrib col_type length = $1;
    attrib col_length length = $5;
    attrib col_format length=$32;
    attrib col_informat length=$32;
    call missing(of _all_);
stop;
run;
proc sql;
 insert into work.MAPPING
 values ('RULEFIRE.Details','RULE_ACTION_FIRE_ID','RULE_ACTION_FIRE_ID','output','1','C','100','','')
 values ('RULEFIRE.Details','RULE_SET_SK','RULE_SET_SK','output','1','N','8','','')
 values ('RULEFIRE.Details','RULE_SET_NM','RULE_SET_NM','output','1','C','100','','')
 values ('RULEFIRE.Details','RULE_SK','RULE_SK','output','1','N','8','','')
 values ('RULEFIRE.Details','RULE_NM','RULE_NM','output','1','C','100','','')
 values ('RULEFIRE.Details','DEPLMT_SK','DEPLMT_SK','output','1','N','8','','')
 values ('RULEFIRE.Details','RULE_FLOW_SK','RULE_FLOW_SK','output','1','N','8','','')
```

```
    values ('RULEFIRE.Details','RULE_FLOW_NM','RULE_FLOW_NM','output','1','C','100','','')
    values ('RULEFIRE.Details','RULE_FIRE_DTTM','RULE_FIRE_DTTM','output','1','N','8','nldatm.',
        'nldatm.')
    values ('RULEFIRE.Details','DEPLMT_EXECUTION_ID','DEPLMT_EXECUTION_ID','output','1','C','100','','')
    values ('RULEFIRE.Details','ENTITY_PRIMARY_KEY','ENTITY_PRIMARY_KEY','output','1','C','1024','','')
    values ('RULEFIRE.Details','TRANSACTION_DTTM','TRANSACTION_DTTM','output','1','N','8','nldatm.',
        'nldatm.')
    values ('RULEFIRE.Details','_RECORDSEQUENCEKEY','_RECORDSEQUENCEKEY','output','1','N','8','','')
    values ('DEPLOY.ThisRun','DEPLMT_SK','DEPLMT_SK','output','2','N','8','','')
    values ('DEPLOY.ThisRun','DEPLMT_NM','DEPLMT_NM','output','2','C','100','','')
    values ('DEPLOY.ThisRun','TRANSACTION_MODE_CD','TRANSACTION_MODE_CD','output','2','C','20','','')
    values ('DEPLOY.ThisRun','RECORDS_PROCESSED_NO','RECORDS_PROCESSED_NO','output','2','N','8','','')
    values ('DEPLOY.ThisRun','TEST_FLG','TEST_FLG','output','2','C','1','','')
    values ('DEPLOY.ThisRun','START_DTTM','START_DTTM','output','2','N','8','nldatm.',' nldatm.')
    values ('DEPLOY.ThisRun','END_DTTM','END_DTTM','output','2','N','8','nldatm.','nldatm.')
    values ('RULEFIRE.Summary','RULE_SK','RULE_SK','output','6','N','8','','')
    values ('RULEFIRE.Summary','RULE_NM','RULE_NM','output','6','C','100','','')
    values ('RULEFIRE.Summary','RULE_SET_SK','RULE_SET_SK','output','6','N','8','','')
    values ('RULEFIRE.Summary','RULE_SET_NM','RULE_SET_NM','output','6','C','100','','')
    values ('RULEFIRE.Summary','RULE_FLOW_SK','RULE_FLOW_SK','output','6','N','8','','')
    values ('RULEFIRE.Summary','RULE_FLOW_NM','RULE_FLOW_NM','output','6','C','100','','')
    values ('RULEFIRE.Summary','ruleFiredCount','ruleFiredCount','output','6','N','8','','')
    values ('INDATA.InData','EngineSize','EngineSize','input','4','N','8','','')
    values ('INDATA.InData','Make','Make','input','4','C','13','','')
    values ('INDATA.InData','Model','Model','input','4','C','40','','')
    values ('INDATA.InData','MSRP','MSRP','input','4','N','8','','')
    values ('INDATA.InData','Type','Type','input','4','C','8','','')
    values ('OUTLIB.MyResults','EngineSize','EngineSize','output','5','N','8','','')
    values ('OUTLIB.MyResults','Make','Make','output','5','C','13','','')
    values ('OUTLIB.MyResults','Model','Model','output','5','C','40','','')
    values ('OUTLIB.MyResults','MSRP','MSRP','output','5','N','8','','')
    values ('OUTLIB.MyResults','Type','Type','output','5','C','8','','')
    ;
quit;
```

Chapter 11
Post-Migration Macros

Overview of Post-migration Macros

After migrating a SAS deployment from one UNIX operating system to another, use the
SAS Decision Manager post-migration macros. The macros ensure that all data, content,
and link and filename references that are used by SAS Decision Manager are accessible
by the new SAS 9.4 deployment. This section describes what action each macro
performs and the syntax that is used by the macros. For more information, see
.

Dictionary

%MM_migrationStep1() Macro

%MM_migrationStep1 macro ports all catalogs and SAS data sets from the source server into .cpo
and .dpo files, and then places them in a user-specified directory.

Requirement: You must run this macro if you are migrating from one UNIX system to a different
type of UNIX system.

Syntax

filename *mycode* catalog "sashelp.modelmgr.migration.source";

%include *mycode*;

filename *mycode*;

%MM_migrationStep1(
Server = *mySourceServer*,
PortNumber = *port*,
ServiceRegistryURL=*SAS-service-registry-URL*,

User = *mmUser,*
Password = *mmPassword,*
TargetDir= *network**port*);

Required Arguments

Server

specifies the server name or multicast address for the migration source server that hosts the model repository. This argument is ignored for a SAS Decision Manager 2.2 or 3.1 to 3.1 migration, which uses the ServiceRegistryURL argument.

Example   `Server=myserver.com`

PortNumber

specifies the port number for the migration source server that hosts the model repository.

Example   `PortNumber=6411`

ServiceRegistryURL

specifies the service registry URL. The host is the server name where the middle-tier is installed and the port is the port number of the SAS Web Application Server.

User

specifies a valid SAS Decision Manager user to access migration source server.

Password

specifies the password for the SAS Decision Manager user to access migration source server.

TargetDir

specifies the directory where all of the portable files are saved.

Example   `TargetDir=\\network1\transfer`

%MM_migrationStep2() Macro

%MM_migrationStep2 macro imports all of .cpo and .dpo files into the target server.

Requirement: You must run this macro if you are migrating from one UNIX system to a different type of UNIX system.

Syntax

filename *mycode* catalog "sashelp.modelmgr.migration.source";

%include *mycode*;

filename *mycode*;

%MM_migrationStep2(ServiceRegistryURL=*service-registry-URL,*
User = *mmUser,*
Password = *mmPassword,*
SourceDir=*network**port,*
TargetDir= *myTargetDIR*);

Required Arguments

ServiceRegistryURL
  specifies the service registry URL. The host is the server name where the middle-tier is installed and the port is the port number of the SAS Web Application Server.

Restriction	This argument is valid only for SAS Decision Manager 2.2 or later.
Example	ServiceRegistryURL= nrstr(http://myServer:80/SASWIPClientAccess/remote/ServiceRegistry)

User
  specifies a valid SAS Decision Manager user to access migration target server.

Password
  specifies the password for the SAS Decision Manager user to access migration target server.

SourceDir
  specifies the directory where all portable files are saved.

Example	SourceDir=\\network\transfer

TargetDir
  specifies the directory where the migration result table is saved. Here is an example:

  TargetDir=\\network1\migration

Index

Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore
for additional books and resources.

§sas
THE POWER TO KNOW®