SAS® Business Rules Manager 2.2

Administrator's Guide

Contents

PART 3   Import and Export Macros   63

Part 1

Introduction to SAS Business Rules Manager

Chapter 1

Overview of SAS Business Rules Manager

About Decision Management

Decision management systems can transform the way businesses make decisions. They enable businesses to use the information they already have to make better decisions—decisions that are based on predictive analytics rather than on past history. Decision management systems automate the process of making decisions, particularly day-to-day operational decisions. They improve the speed, efficiency, and accuracy of routine business processes, in part by reducing the need for human intervention. By automating decisions, organizations in every industry can improve interactions with customers, partners, suppliers, and employees. In addition, organizations that are highly regulated, such as financial services, health care, and insurance, can more easily achieve compliance as a result of repeatable, traceable decisions.

About SAS Business Rules Manager

Business rules capture the logic of business decisions and are one of the core components of decision management systems. Business rules make the decision-making process transparent and adaptable, allowing organizations to respond quickly to new information about customers and markets. They allow organizations to identify and deal with fraud, avoid unnecessary risk, and find opportunities hidden in customer data.

You can use SAS Business Rules Manager to create a database of business rules, connect those rules together into rules flows, and publish the rule flows for use by other applications. SAS Business Rules Manager provides the following capabilities:

data management
> You can manage your list of data tables from within the application. You can create new Base SAS libraries, add and remove tables, view table data and metadata, create and delete table summaries, and associate attachments and comments with tables. The application uses these data tables whenever it needs to access data, such as for rule discovery and rule flow testing.

vocabulary management
> A business vocabulary identifies the objects and actors in your business domain. It defines the entities and terms that are the building blocks of business rules. SAS Business Rules Manager enables you to easily create and edit entities and terms. For individual terms, you can create a list of allowable values, which makes creating rules even easier.

business rule authoring
> A business rule specifies conditions to be evaluated and action to be taken if those conditions are satisfied. For example, you can create a rule that determines whether a customer has a mortgage. That same rule can then add the outstanding balance of the mortgage to a running total of the customer's debt. With SAS Business Rules Manager, you define the conditions and actions for each rule. You can use the Expression Editor to create the expressions for the rule.

rule set management
> A rule set is a logical collection of rules. A single rule set can have many rules. For example, you might have a rule set that determines a customer's asset balance and another rule set that determines a customer's debt level. SAS Business Rules Manager displays rules sets in decision tables. Each row of the decision table defines the conditions and actions for one rule. By using SAS Business Rules Manager, you can easily create new rule sets, reorder the rules in a rule set, add new rules to existing rule sets, and more.

rule flow authoring and publishing
> A rule flow is a logical collection of rule sets. A rule flow defines a set of rule sets and the order in which they must be executed. A single rule flow frequently corresponds to a single decision. For example, a rule flow can initially execute the rule set that determines a customer's asset balance. Next, the rule set that determine a customer's debt level is executed. Finally, the rule set that assign's a customer's loan application status is executed.
>
> SAS Business Rules Manager makes it easy to combine rules sets into a rule flow and to publish those rule flows to the metadata server. After a rule flow has been published, it is available for use by other applications.

Viewing Help and Documentation

SAS Business Rules Manager provides the following types of Help and documentation:

how-to Help
> How-to Help provides quick instructions or tips to help you complete some tasks in the application. To access how-to Help, select **Help ⇨ How To**.

embedded Help
> Help pop-up menus and tooltips provide brief descriptions of various fields.
>
> To access a Help pop-up menu for a field, click the Help icon (②) when it appears next to a field. You can also place the mouse pointer over an element in the SAS Business Rules Manager windows to view the associated tooltip.

SAS Business Rules Manager: User's Guide
> This document provides detailed information about the concepts and tasks that are related to using SAS Business Rules Manager. This document is available at http://support.sas.com/documentation/onlinedoc/brm.

SAS Business Rules Manager: Administrator's Guide
> This document contains information about the administration tasks that are required
> to set up and configure the SAS Business Rules Manager and is available at http://
> support.sas.com/documentation/onlinedoc/brm.

Additional resources are available from the **Help** menu. To access these resources, select
Help ⇨ **SAS on the Web**.

Chapter 2

Architecture for System Administrators

Architecture of the SAS Intelligence Platform

The SAS Intelligence Platform is designed to efficiently access large amounts of data, while simultaneously providing timely intelligence to a large number of users. SAS Intelligence Platform uses an *n*-tier architecture that enables you to distribute functionality across computer resources, so that each type of work is performed by the resources that are most suitable for the job.

A *tier* in the SAS architecture represents a conceptual category of software components that perform similar types of computing tasks and that require similar types of resources. Different tiers do not necessarily represent separate computers or groups of computers. More than one computer can be used for a specified tier as well.

You can modify the SAS architecture to meet the demands of your workload. For a large company, the architecture can easily consist of many computers with different operating environments. For prototyping, demonstrations, or very small enterprises, the components for all of the tiers can be installed on a single computer.

The architecture of the SAS Intelligence Platform consists of the following tiers.

Client	The client tier provides users with access to intelligence data and to functionality through web-based interfaces.
Middle	The middle tier provides an environment in which the SAS Business Rules Manager web application can execute. The middle-tier applications rely on servers on the SAS server tier to perform SAS processing, including data query and analysis.

SAS Server	The SAS server tier includes the SAS Metadata Server and several compute servers that execute SAS analytical and reporting processes for distributed clients.
	The compute servers include the SAS OLAP Server, the SAS Workspace Server, the SAS Pooled Workspace Server, and the SAS Stored Process Server. The SAS object spawner process runs on server machines and handles requests for SAS Workspace Servers, SAS Pooled Workspace Servers, and SAS Stored Process Servers.
Data	The data tier stores your enterprise data. All of your existing data assets can be used, whether your data is stored in relational database management systems, SAS tables, or enterprise resource planning system (ERP) tables.

For more information about the SAS Intelligence Platform, see *SAS Intelligence Platform: Overview* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

The SAS Intelligence Platform and SAS Business Rules Manager

The SAS Intelligence Platform architecture is a comprehensive, end-to-end infrastructure for creating, managing, and distributing enterprise intelligence. This architecture consists of the following tiers:

client tier
> provides users with desktop access to data and functionality through easy-to-use interfaces. With SAS Business Rules Manager, users author rule sets and rule flows through the SAS Business Rules Manager Flex client.

middle tier
> provides web-based interfaces for creating reports and distributing information, while passing analysis and processing requests to the SAS servers. The middle tier provides SAS Business Rules Manager services.

server tier
> provides SAS servers that process data and handle client requests. For SAS Business Rules Manager, the server tier provides the SAS Business Rules Engine.

data tier
> stores your data. The business rules database contains all of the data that the user enters through the SAS Business Rules Manager Flex client application.

The following figure shows how SAS Business Rules Manager is deployed on the SAS Intelligence Platform.

Figure 2.1 *SAS Intelligence Platform Architecture and SAS Business Rules Manager*
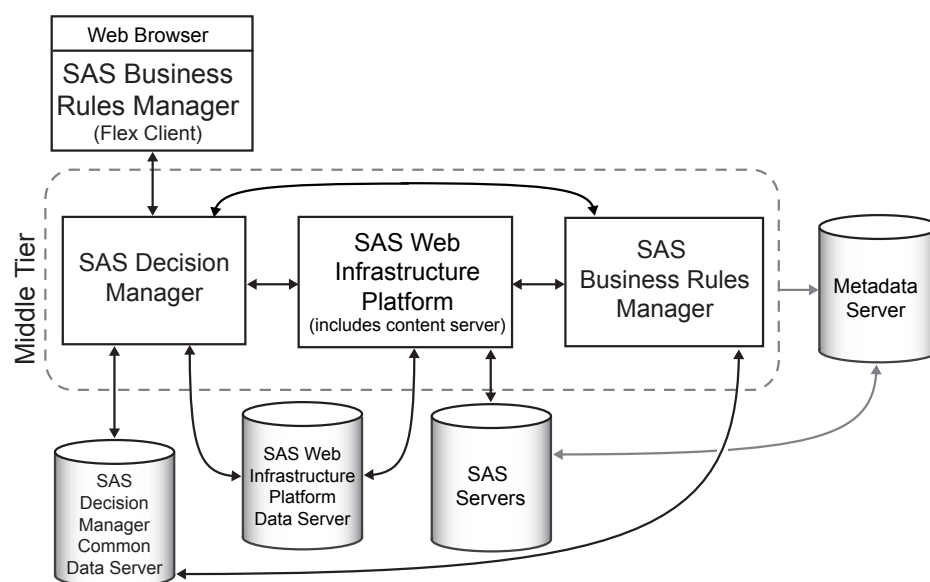
SAS Business Rules Manager
> the Flex client application for SAS Business Rules Manager. Through this client application, users author vocabularies, rule sets, and rule flows.

SAS Decision Manager
> manages requests to SAS Business Rules Manager and provides general SAS Decision Manager functions such as data source and workflow management.

SAS Web Infrastructure Platform
> provides common SAS infrastructure services that SAS Business Rules Manager uses to authenticate users and to access services within the SAS platform.

SAS Business Rules Manager on the middle tier
> manages communication with the business rules database and initiates the process of saving rule flows to the content repository.

SAS Web Infrastructure Platform Data Server
> serves as transactional storage for SAS middle-tier software and some SAS solutions software.

SAS Servers
> SAS application servers that execute SAS code that is submitted from the middle-tier applications.

SAS Decision Manager Common Data Server
> contains all of the data that users enter through the SAS Business Rules Manager client.

Metadata server
> contains the BusinessRuleFlow public metadata objects that are created when a rule flow is published. These objects are used by the integrated SAS applications to execute rule flows.

Process for Publishing Rule Flows

When a user publishes a rule flow, SAS Business Rules Manager creates an XML file and a BusinessRuleFlow metadata object. The XML file is stored in the content repository, and the metadata object is stored on the metadata server.

The following figure illustrates the process of publishing rule flows.

Figure 2.2 *Process for Publishing Rule Flows*
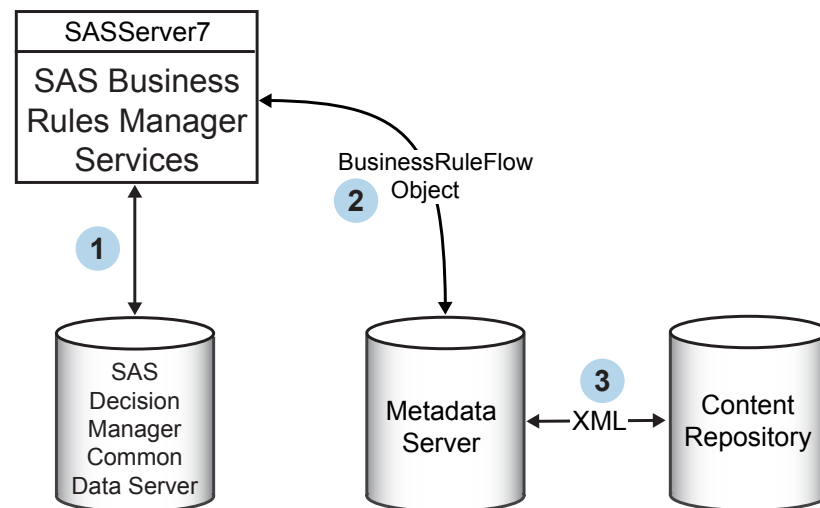
1 SAS Business Rules Manager reads the rule flow data in the business rules database.

2 SAS Business Rules Manager generates the BusinessRuleFlow public metadata object and the XML file. SAS Business Rules Manager stores the metadata object on the metadata server.

3 The metadata object stores the XML file in the content repository.

Part 2

Installation and Configuration

Chapter 3
Pre-Installation Tasks for SAS Business Rules Manager

About the Pre-Installation Tasks

Before you begin to install SAS Business Rules 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 pre-installation tasks before you install SAS Business Rules Manager:

1. Verify that your system meets the minimum requirements.

2. Determine the location of the SAS environment file.

3. Determine the database that you want to use.

4. Determine whether you need to synchronize the time zones that are specified in all of your operating environments.

5. Install the prerequisite software.

6. Create standard user accounts in the operating system.

7. Obtain a deployment plan and installation data file.

8. Download your software and create a software depot.

The following topics provide details about each step.

Verify Operating System Requirements

Ensure that you meet the minimum requirements that are described in the appropriate system requirements document for your installation. System requirements are unique for each operating system. They include software requirements, hardware requirements, space requirements, specific product requirements, and graphics hardware and software compatibility. System requirements documentation is available at http://support.sas.com/documentation/installcenter/.

Determine the Location of the SAS Environment File

During deployment of SAS Business Rules Manager, you are prompted by the SAS Deployment Wizard to specify the location of the SAS environment file (named sas-environment.xml). An example is http://*<server>:<port>*/sas/sas-environment.xml). This file defines a set of SAS deployments at your site for client applications to use. The sas-environment.xml file does not need to physically exist at the location that you specify in the SAS Deployment Wizard before beginning the SAS installation. However, knowing the intended location of this file is important because every client installation is prompted for this value. If you do not specify the correct URL for the SAS environment file during deployment, then you must manually specify the URL in a file for every client. The URL is commonly stored in this location: **\\<server>\\<drive>: \\Program Files\\SASHome\\sassw.conf**. As a best practice during your planning process, determine a URL and share it with administrators who perform installations.

For more information about the structure of this file, see Appendix 1, "Configuring the SAS Environment File," in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Determine the Database to Use

You can use either Oracle or the SAS Decision Manager Common Data Server for your 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 Chapter 2 of *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 17. 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 16.

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 be deploying 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.

Install the Prerequisite Software

Before you install SAS Business Rules Manager, install the following prerequisite software:

• Oracle, if you are using Oracle for your database. See "Pre-Installation Tasks for an Oracle Database" on page 17 for instructions.

• Adobe Flash Player version 10.1.0 or later. This software is required on each client machine that accesses SAS Business Rules Manager.

Create Standard User Accounts

As a pre-installation task, you must create the following user accounts in the operating system:

• an account for the user who will install and configure the SAS software

• an account to run the spawned SAS servers

You should also create a SAS Server Users group on Windows and a sas group on UNIX.

For important details about setting up these users and groups, see the pre-installation checklist for your deployment. Also see Chapter 2, "Setting Up Users, Groups, and Ports," in *SAS Intelligence Platform: Installation and Configuration Guide* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

Obtain a Deployment Plan and a SAS Installation Data File

Before you can install your SAS software, you must obtain a deployment plan. The deployment plan is an XML file that specifies the software that you will install and configure on each machine in your environment. The plan serves as input to the SAS Deployment Wizard. A deployment plan can be a custom plan for your specific software installation, or it can be a standard, predefined plan that describes a common configuration. For more information, see "About Deployment Plans" in Chapter 6 of *SAS Intelligence Platform: Installation and Configuration Guide*.

You must also obtain a SAS Installation Data (SID) file. The SID file contains license (SETINIT) information that is required to install SAS.

Note: When you create a deployment plan that includes SAS Business Rules Manager, you will see SAS Decision Manager components in your plan even if you have not licensed SAS Decision Manager or included the SAS Decision Manager offering in your plan.

Download Your Software and Create a SAS Software Depot

Use the SAS Download Manager to download the software that is listed in your SAS Software Order. The SAS Download Manager creates a SAS Software Depot from which you install your software. For more information, see Chapter 3, "Creating a SAS Software Depot," in *SAS Intelligence Platform: Installation and Configuration Guide*. You can then use the SAS Deployment Wizard to install your software.

Pre-Installation Tasks for SAS Decision Manager Common Data Server

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

The following table describes the information that you need in order to complete the steps in the SAS Deployment Wizard.

Table 3.1   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**.

Property	Description
Database User	Specifies the user name for the database administrator. This user owns the database and has superuser privileges.
Database Password	Specifies a password for the user name 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.
Database Name	Specifies the database name. The default name for the database is `dcmdb`. *Note:* The PostgreSQL database type is case sensitive.
User ID	Specifies the user name for the user whose credentials will be used to access the SAS Decision Manager Common Data Server database.

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 Common Data Server database, perform the following steps before you install SAS Business Rules Manager:

1. Install the Oracle database server.
2. Install the JDBC drivers.
3. Install a database client application.
4. Determine the required database information.
5. Specify the required database privileges.
6. Test the connection to your database.

Install the Oracle Database Server

If you are using Oracle for your database, then you must install an Oracle database server. You must install this third-party software before you install SAS Business Rules Manager. For more information, see the system requirements documents listed in "Verify Operating System Requirements" on page 14.

Install JDBC Drivers for Oracle

You must download the following JDBC drivers and place them in a separate directory that does not contain any other files on all middle-tier servers to ensure proper installation and configuration of SAS Business Rules Manager.

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 listed in "Verify Operating System Requirements" on page 14.

Install the Oracle Client Application

You must install and configure an Oracle client application on all server-tier machines. If you choose to bypass database initialization when you run the SAS Deployment Wizard, you must run a set of database scripts to prepare and initialize your database. For information about these scripts, see "Create Oracle Database Tables" on page 31.

Also, you must have an entry in the tnsnames.ora file for that Oracle client that corresponds to the database that you have set up.

Determine the Information Required for the Oracle Database

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

The following table describes the information that you need in order to complete the steps in the SAS Deployment Wizard.

Table 3.2 SAS Deployment Wizard Information for Oracle

Property	Description
Database Type	Specifies the database type to use for the SAS Decision Manager database. Select **Oracle**.
User Name	Specifies the user name for the database that is used with your SAS Business Rules Manager installation.
Password	Specifies a valid password for the user name associated with the database account.
Port	Specifies the port that is used by the database. The default port for Oracle is 1521.
Host Name	Specifies the fully qualified host name of the server on which the database is installed.

Property	Description

Database Name	Specifies the Oracle database name. The **Net Service Name** and the **Service Name** fields that are configured in the tnsnames.ora file must be the same. You must use this value for the **Database Name** field in the SAS Deployment Wizard. For example, if you had the following entry in the tnsnames.ora file, you would enter **monitordb** in the **Database Name** field in the SAS Deployment Wizard:
	
	``monitordb =
	  (DESCRIPTION =
	    (ADDRESS_LIST =
	      (ADDRESS =
	        (COMMUNITY = TCP_COMM)
	        (PROTOCOL = TCP)
	        (HOST = hostname.your.company.com)
	        (PORT = 1521)
	      )
	    )
	    (CONNECT_DATA =
	      (SERVICE_NAME = monitordb)
	    )
	  )```
	
	The **Net Service Name** and **Service Name** in this example are the same.

Schema	Specifies the schema name for the database. The default schema is the same as the user ID.
DBMS JAR File	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 Business Rules Manager in order to configure SAS Business Rules Manager Common Data Server. See "Install JDBC Drivers for Oracle" on page 18 for more information.

Specify the Required Database Privileges for Oracle

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

- 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.

Chapter 4
Installing SAS Business Rules Manager

About the SAS Deployment Wizard

You use the SAS Deployment Wizard to install and configure the SAS software and related products that are included in your deployment plan file. When you execute the SAS Deployment Wizard, you select the deployment type that you are performing. You can install and configure the software in a single execution of the wizard, or you can install and configure the software in two separate executions. The latter approach gives you the opportunity to test the SAS license before the configuration step.

The SAS Deployment Wizard prompts you to perform a variety of tasks, including the following items:

- specify the software order, the deployment plan, and the SAS software products that you are installing and configuring

- specify host machine information

- specify information about user accounts that were created in the pre-installation phase

- for multiple-machine configurations, install the server-tier, middle-tier, and client-tier software on the appropriate machines

For more information, see Chapter 6, "Installing and Configuring Your SAS Software," in *SAS Intelligence Platform: Installation and Configuration Guide*.

Note: The SAS Deployment Wizard prompts you for SAS Decision Manager settings even if you have not licensed SAS Decision Manager or included the SAS Decision Manager offering in your plan.

Single-Machine versus Multiple-Machine Installations

You can install SAS Business Rules Manager on one or on several machines. This choice is determined when you order SAS Business Rules Manager and is detailed in the deployment plan XML file.

For multiple-machine installations, you must first install SAS Business Rules Manager on the server-tier machine. You can then install SAS Business Rules Manager on other additional machines that are part of a middle tier in your configuration. For guidelines on installing SAS on multiple machines, see "Installation Order Rules for Multiple Machine Deployments" in Chapter 6 of *SAS Intelligence Platform: Installation and Configuration Guide*.

The server tier consists of a set of SAS servers that are installed as a part of the SAS Intelligence Platform. The server tier contains the code generation macro that is necessary for executing rules and integrating SAS Business Rules Manager with other SAS products.

Products Installed with SAS Business Rules Manager

Your deployment plan for SAS Business Rules Manager includes additional SAS products that support and complement SAS Business Rules 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 Chapter 6 of *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/en/ikdeploywizug/66034/PDF/default/user.pdf.

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 Business Rules 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 Business Rules 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 Business Rules 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 Common Data Server database. For information about which versions of the alternative databases are supported, see "Reviewing Third-Party Database Requirements" in Chapter 6 of *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 16 and "Pre-Installation Tasks for an Oracle Database" on page 17 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 SAS Decision Manager Common Data Server 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 31 for more information.

Chapter 5

Post-Installation Tasks for SAS Business Rules Manager

Overview of Post-Installation Tasks

After you install SAS Business Rules Manager, complete the following post-installation tasks:

1. Follow the instructions in the Instructions.html file.

2. Create users and assign permissions.

3. Create an operating system account for product administrators and users.

4. (Optional) Create operating system accounts in UNIX environments. on page 28

5. (Optional) Run a database script to create Oracle synonyms in the database.

6. (Optional) Create Oracle database tables.

7. Verify that the Certificate Authority certificate is available.

8. (Optional) Set up the UUID generator daemon.

9. ReviewSAS Business Rules Manager Web properties in SAS Management Console.

10. Configure SAS Workflow for use with SAS Business Rules Manager. For more information, see "Configuring SAS Workflow for Use with SAS Business Rules Manager" on page 57.

11. (Optional) Modify log file settings.

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

Follow Instructions in Instructions.html

At the end of the installation process for SAS Business Rules Manager, the SAS Deployment Wizard produces an HTML document named Instructions.html. If your server tier and middle tier are hosted on separate machines, there is an Instructions.html file for each machine.

The Instructions.html file is located in ***sasconfigdir*\\Lev#\\Documents**. Follow the instructions that are provided in the HTML documents.

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.

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 27.

In a UNIX environment, all SAS Business Rules Manager users must be part of a group that has the appropriate group permissions.

For more information, see "Creating Operating System Accounts in UNIX Environments" on page 28 and Chapter 8, "Configuring Users, Groups, and Roles," on page 49.

Create an Operating System Account for Product Administrators and Users

About the User Accounts for SAS Business Rules Manager

SAS Business Rules Manager provides two types of user accounts:

Product administrator
A SAS Business Rules Manager administrative user is specific to SAS Business Rules 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.

You must create the operating system account for the administrator as post-installation task. For more information, see "Create Windows Operating System Accounts and Groups for Users" on page 27.

Users of SAS Business Rules 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 can create regular user accounts for SAS Business Rules Manager as a post-installation task. For more information, see Chapter 8, "Configuring Users, Groups, and Roles," on page 49.

Create Windows Operating System Accounts and Groups for Users

On the SAS Workspace Server, create an operating system account for the administrator of SAS Business Rules Manager (for example, mdlmgradmin) and all SAS Business Rules Manager users.

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

- If you are working on a local machine, complete these steps to create this user account:

  1. If you are running in a Windows operating system environment, 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**.

- Define the user (for example, *<domain>\username*) on the Active Directory server.

Create UNIX Operating System Accounts and Groups for Users

You can create the SAS Business Rules Manager UNIX user group as a pre-installation or post-installation task. For more information, see .

Creating Operating System Accounts in UNIX Environments

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 Business Rules Manager, you can assign all users to the same group and grant the same permissions to all users at one time. All SAS Business Rules 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.

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 Business Rules Manager in a SAS session.

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

- The SAS scripts are updated to grant permissions to the SAS Business Rules Manager users on the SAS Workspace Server. For more information, see "Update the SAS Scripts to Grant Permissions to the User Group" on page 29.

- 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 Business Rules 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.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
   ```

   a 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.

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

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

   Here are code examples for each UNIX environment where SAS Business Rules 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```

Create Oracle Database Synonyms

If you use Oracle for your SAS Decision Manager Common Data Server 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 **SAS_HOME\SASDecisionManagerCommonMidTierforDecisionManager\2.2 \Config\Deployment\dbscript\Oracle\optional**. Use your preferred Oracle tool to run these scripts. One of these scripts, **brm_oracle_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_oracle_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_oracle_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_oracle_create_synonym.sql** using your preferred Oracle tool. This script does not require substitution variables.

4. Run **brm_oracle_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 Common Data Server. (See "Create and Load Tables through the SAS Deployment Wizard" on page 23.) 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 following scripts to create the business rules tables with a compatible database client for your installation. These scripts are located in **SAS_HOME\SASDecisionManagerCommonMidTierforDecisionManager\ 2.2\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

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 Chapter 5, "Setting Up Certificates for SAS Deployment," in *SAS Intelligence Platform: Installation and Configuration Guide*.

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 Chapter 19 of *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 using the SAS Deployment Wizard, additional steps are required 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 Chapter 19 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Set Up the UUID Generator Daemon

The business rules engine uses the UUIDGEN function to create unique identifiers for rule-fired records. Unique identifiers are necessary for merging rule-fired data sets. If you are executing rules in a UNIX operating environment, you need to set up the object spawner to be a UUID Generator Daemon (UUIDGEND). For information, see "Universal Unique Identifiers and the Object Spawner " in Chapter 39 of *SAS Language Reference: Concepts*.

In addition, you should specify the UUIDGENHOST system option for any jobs that run code that was generated by the business rules engine. For more information, see "UUIDGENDHOST= System Option" in *SAS System Options: Reference*.

Review Business Rules Manager Web Properties

Review the Business Rules Manager Web 2.2 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**.

2. Right-click **Business Rules Manager Web 2.2** and select **Properties**. The **Business Rules Manager Web 2.2 Properties** dialog box appears.

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 applies only to SAS Business Rules Manager 2.1. It is not used by SAS Business Rules Manager 2.2.

Max row count per table
the maximum number of rows per rule flow test table. This property applies only to SAS Business Rules Manager 2.1. It is not used by SAS Business Rules Manager 2.2.

Temporary test code generation directory
a temporary directory that SAS Business Rules Manager uses while it generates SAS code for rule flow tests.

Largest allowed uploaded lookup table row count
the maximum number of rows that can be uploaded for a lookup table.

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 Business Rules Manager uses while it generates the SAS code for rule sets and rule flows.

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.

Test Metadata Library Root Directory
the directory in which rule flow test metadata is stored.

Modify Log File Settings

Log4j Configuration File

SAS Business Rules Manager uses log4j to perform logging. When SAS Business Rules Manager starts, the log4j configuration file for the web application is read from **`SAS-config-dir\Lev1\Web\Common\LogConfig \SASBusinessRulesManagerWeb-log4j.xml`**. This file is a standard log4j configuration file.

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 5.1 Logging Priority Levels

Priority	Description
DEBUG	The most verbose logging level. This level displays information that is most useful for debugging an application. SAS Business Rules 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 Business Rules Manager.
INFO	Verbose logging level. This level displays messages that highlight the progress of an application. SAS Business Rules 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 Business Rules 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 Business Rules 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 Business Rules Manager.

Log Files

SAS Business Rules Manager writes information to the following log files:

SASBusinessRulesManagerWeb2.2.log
    contains messages from SAS Business Rules Manager

SASDecMgrCommon2.2.log
    contains messages from the Workflow and Data plug-ins

SASDecMgrShell2.2.log
    contains general messages from the Shell

By default, SAS Business Rules Manager writes log files to
SAS-config-dir\Lev1\Web\Logs\SASServer7_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 Chapter 8 of *SAS Intelligence Platform: Middle-Tier
Administration Guide* for more information about this configuration file.

SAS Business Rules Manager creates new log files each day. For information about
logging configurations, see "Modifying Your Server Logging Configurations" in Chapter
10 of *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>
```

Directories for Business Rules Metadata and XML Files

SAS Business Rules Manager creates two directories for business rules metadata: **Products** and **/System**. These directories are on the content server.

SAS Business Rules 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. For more information, see .

For content that was published with an earlier release of SAS Business Rules Manager, the current release of SAS Business Rules Manager continues to use the original publish location. Do not delete directories created by earlier releases of SAS Business Rules Manager.

Delete XML Content from the SAS Decision Manager Database

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.

Chapter 6
Migrating to SAS Business Rules Manager 2.2

About the SAS Business Rules Manager Migration Process

The SAS Business Rules Manager 2.2 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 business rules 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 Tasks

For information about the pre-migration tasks that you must perform, see Chapter 3, "Performing Pre-migration Tasks," in *SAS Intelligence Platform: Migration Guide*. Here are the most important steps:

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

• Back up the SAS Decision Manager Common Server database if you are migrating from SAS Business Rules Manager 2.2 to 2.2 on a SAS 9.4 system.

- Perform any required maintenance that is required to meet minimum baselines.

- Determine whether you can use a standard deployment plan to install SAS 9.4 and SAS Business Rules Manager or whether you need to request a custom deployment plan. You must specify the plan file when you use the SAS Deployment Wizard.

- Complete the pre-installation and migration checklists. These checklists can be customized based on the deployment plan that you choose. For more information, see "Completing the Pre-migration Checklists" in Chapter 3 of *SAS Intelligence Platform: Migration Guide*.

- If you are moving to a new system, ensure that the required operating system user accounts that you used for SAS in your current operating system also exist in your new operating system. Examples of these user accounts include **sasadm**, **sastrust**, and **sasdemo**.

- Before you migrate to SAS Business Rules Manager 2.2, you should record the database settings in your current environment. You must enter this information in SAS Deployment Wizard.

  - If you are using the SAS Web Infrastructure Platform Data Server for your SAS Business Rules Manager 2.1 database, record the database name and the user name for your database. The default database name is **brmdb**.

  - For the SAS Decision Manager Common Data Server, record the database name and the user name for your database. The default database name is **dcmdb**.

  - If you are using Oracle for your database, record the following information:

    - fully qualified host name of the database server

    - port number of the database server

    - Oracle Site Identifier (SID)

    - user ID of the database user whose credentials are used to access SAS Business Rules Manager data on the server

    - password of the database user whose credentials are used to access SAS Business Rules Manager data on the server

    You can find the Oracle SID in the tnsnames.ora file. You can also determine the SID by running the following query using a database user ID on your Oracle instance:

    ```
    select instance from v$thread
    ```

    You must enter the SID in the **Database name** field in the SAS Decision Manager Database JDBC Properties dialog box in SAS Deployment Wizard.

- Install third-party software. 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 for the SAS Decision Manager Common Data Server database. See for information about the tnsnames.ora file.

- Create a SAS Software Depot. For more information, see "Creating SAS Software Depots" in Chapter 3 of *SAS Intelligence Platform: Migration Guide*.

- Use the SAS Migration Utility to create a migration package.

  For more information, see Appendix 1, "SAS Migration Utility Reference," in *SAS Intelligence Platform: Migration Guide*.

Post-Migration Tasks

About Running the Migration Scripts

To migrate to SAS Business Rules Manager 2.2, you must run migration scripts as part of the SAS post-migration process. These scripts update the structure of the database tables, and add new tables, sequences, and triggers that are new with SAS Business Rules Manager 2.2.

Note: In a multi-tier environment, copy the migration scripts to the server tier before you run them.

Migrate from SAS Business Rules Manager 1.2 to 2.2 for Oracle

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

!SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/2.2 /Config/Deployment/dbscript/oracle/migration/

To migrate an Oracle database from SAS Business Rules Manager 1.2 to 2.2:

1. Edit the **migration_brm_1.2m1_to_brm_2.2_sql** migration script. Replace **@schema.name@** with the schema name for your database.

2. Run the **migration_brm_1.2m1_to_brm_2.2_sql** migration script.

3. Review the Business Rules Manager Web 2.2 properties on page 32 in SAS Management Console to ensure that the values are appropriate for your environment.

4. In SAS Management Console, add all users that need access to SAS Business Rules Manager to the Decision Manager Users group.

5. Update your user group memberships as needed. See "Update User Group Membership" on page 42 for more information.

6. Sign in to SAS Business Rules Manager and republish rule flows if needed. You must republish rule flows that were published with SAS Business Rules Manager 1.2 in order for them to be valid in a SAS Data Integration Studio job. The previous version can no longer be referenced through SAS Data Integration Studio, and the rule flow cannot be edited in SAS Business Rules Manager 2.2. However, scheduled jobs will continue to work.

Migrate from SAS Business Rules Manager 2.1 to 2.2 for Oracle

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

!SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/2.2 /Config/Deployment/dbscript/oracle/migration/

To migrate an Oracle database from SAS Business Rules Manager 2.1 to 2.2:

1. Edit the **migration_brm_2.1_to_brm_2.2_sql** migration script. Replace **@schema.name@** in each file with the schema name for your database.

2. Run the **migration_brm_2.1_to_brm_2.2_sql** migration script.

3. 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 32 for more information.

4. Update your user group memberships as needed. See "Update User Group Membership" on page 42 for more information.

Migrate from SAS Business Rules Manager 2.1 to 2.2 for SAS Decision Manager Data Server

You need to run PostgreSQL commands to migrate and upgrade the database. These commands are located in the following directory:

!SASHOME/SASWebInfrastructurePlatformDataServer/9.4/bin

When you run these commands, substitute the correct values for the host name, port number, and user ID for your database. These commands prompt you to enter a password.

Note: If you are performing the migration in a UNIX operating environment and these commands fail, you might need to set environment variables. See "Set Environment Variables for UNIX" on page 43 for more information.

To migrate to SAS Business Rules Manager 2.2 and use the SAS Decision Manager Data Server:

1. On the source machine, run the **pg_dump** command. This command creates a script file that contains the SQL commands that are needed to reconstruct the database:

   ```
   pg_dump --host=host_name --port=nnnn --username=database_owner_userID
   --password -Fp -O brmdb > brmdb.dump
   ```

   The –O option exports the database without commands that set ownership of any of the database objects.

2. Copy the generated file brmdb.dump to the target server.

3. Stop SASServer7.

4. If SAS Model Manager 13.1 is installed on the target server, run the following command to create a backup of the SAS Model Manager tables:

   ```
   pg_dump --host=host_name --port=nnnn --username=userID
   --password -t sas_mm* database_name > mmbackup-131.dump
   ```

   Note: Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

5. On the target server, run the **dropdb** command to drop the database that was created by the SAS Deployment Wizard:

   ```
   dropdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

6. On the target server, run the **createdb** command to create a new, empty database:

   ```
   createdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

7. On the target server, restore the exported database:

   ```
   psql --host=host_name --port=nnnn -f brmdb.dump database_name userID
   ```

> *Note:* Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

> *Note:* You might see messages such as **no privileges were granted for "public"** or **role *userID* does not exist**. These messages mean that these users are no longer applicable to the upgraded database instance and can be ignored.

8. Run the **migration_brm_2.1_to_brm_2.2_sql** migration script that is located in the following directory:

   !SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/ 2.2/Config/Deployment/dbscript/postgres/migration

9. Run the **edm_workflow_interface_create_*.sql** scripts. These scripts are located in the following directory:

   !SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/ 2.2/Config/Deployment/dbscript/postgres

   > *Note:* In a multi-tier environment, copy the scripts to the server tier before you run them.

   Run the following scripts in the order listed:

   1. edm_workflow_interface_create_table.sql

   2. edm_workflow_interface_create_sequence.sql

   3. edm_workflow_interface_create_constraint.sql

   4. edm_workflow_interface_create_trigger.sql

   5. edm_workflow_interface_required_inserts.sql

10. Restore the SAS Model Manager tables by using the backup that you created in Step 4:

    ```
    psql --host=host_name --port=nnnn -f mmbackup-131.dump database_name userID
    ```

    > *Note:* Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

11. 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 32 for more information.

12. Update your user group memberships as needed. See "Update User Group Membership" on page 42 for more information.

Migrate from SAS Business Rules Manager 2.2 to Version 2.2 (Hardware Upgrade)

You need to run PostgreSQL commands in order to migrate the SAS Decision Manager Common Data Server database. These commands are located in the following directory:

!SASHOME/SASWebInfrastructurePlatformDataServer/9.4/bin

When you run these commands, substitute the correct values for the host name, port number, and user ID for your database. These commands prompt you to enter a password.

Note: If you are performing the upgrade in a UNIX operating environment and these commands fail, you might need to set environment variables. See "Set Environment Variables for UNIX" on page 43 for more information.

1. On the source machine, run the `pg_dump` command. This command creates a script file that contains the SQL commands that are needed to reconstruct the database:

   ```
   pg_dump --host=host_name --port=nnnn --username=database_owner_userID
   --password -Fp dcmdb > dcmdb.dump
   ```

2. Copy the generated file dcmdb.dump to the target server.

3. Stop SASServer7.

4. On the target server, run the `dropdb` command to drop the database that was created by the SAS Deployment Wizard:

   ```
   dropdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

5. On the target server, run the `createdb` command to create a new, empty database:

   ```
   createdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

6. On the target server, restore the exported database:

   ```
   psql --host=host_name --port=nnnn -f dcmdb.dump database_name userID
   ```

   Note: Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

   Note: You might see messages such as `no privileges were granted for "public"` or `role userID does not exist`. These messages mean that these users are no longer applicable to the upgraded database instance and can be ignored.

Update User Group Membership

In previous versions of SAS Business Rules Manager, all users that needed access to SAS Business Rules Manager were assigned to the Business Rules Manager Users group. In SAS Business Rules Manager 2.2, users are assigned to the Decision Manager Users group.

You can continue to use the Business Rules Manager Users group as your primary group, or you can start using the Decision Manager Users group. If you continue to use the Business Rules Manager Users group, you can preserve customized authorization settings.

To continue to use the Business Rules Manager Users group:

1. Ensure that all users that need access to SAS Business Rules Manager are assigned to the Business Rules Manager Users group.

2. If you are using a different database instance in the migrated environment, update the authentication domains for migrated user groups. Complete the following steps in SAS Management Console:

   a. Remove the identity for the authentication domain `edm_db_auth` from the Decision Manager Users group.

    b. Update the identity for the authentication domain **edm_db_auth** for the Business Rules Manager Users group to match the updated login that was added to the Decision Manager Users group.

To start using the Decision Manager Users group as your primary group:

1. In SAS Management Console, assign all users that need access to SAS Business Rules Manager to the Decision Manager Users group.

2. Log on to the SAS Content Server, and update the permissions on all of the directories under **sasdav/Products/SAS Business Rules Manager**. Give members of the Decision Manager Users group recursive Read and Write access.

Set Environment Variables for UNIX

If you are upgrading SAS Business Rules Manager in a UNIX operating environment, you might need to set environment variables in order for the migrate commands described in to succeed. If the migrate commands fail, use the following export commands to set the LD_LIBRARY_PATH and LIBPATH environment variables:

```
export LD_LIBRARY_PATH=/install/cfgsas1/SASHome/
SASWebInfrastructurePlatformDataServer/9.4/lib:$LD_LIBRARY_PATH

export LIBPATH=/install/cfgsas1/SASHome/
SASWebInfrastructurePlatformDataServer/9.4/lib:$LIBPATH
```

Replace *SASHome* with the value for your site.

Chapter 7
Performing an Upgrade-In-Place

Perform an Upgrade-In-Place

When you are upgrading from SAS Business Rules Manager 2.1 on SAS 9.4 to SAS
Business Rules Manager 2.2 on the second maintenance release of SAS 9.4, you can
perform an upgrade-in-place.

To perform an upgrade-in-place:

1. Complete the pre-upgrade tasks for the database that you are using.

2. Perform an upgrade-in-place. For more information, see *SAS Guide to Software
   Updates* at http://support.sas.com/documentation/whatsnew/index.html.

3. Follow the instructions in the UpdateInstructions.html file. At the end of the upgrade
   process, the SAS Deployment Wizard produces an HTML document named
   UpdateInstructions.html. If your server tier and middle tier are hosted on separate
   machines, each machine has an UpdateInstructions.html file. Follow the instructions
   that are provided in the HTML documents.

4. Complete the post-upgrade steps for the database that you are using. For SAS Web
   Infrastructure Platform Data Server, see "Post-Upgrade Steps for SAS Web
   Infrastructure Platform Data Server" on page 46 for instructions. If you are using
   Oracle for your SAS Decision Manager database, the post-upgrade steps are the
   same as the post-migration steps. For more information, see "Migrate from SAS
   Business Rules Manager 2.1 to 2.2 for Oracle" on page 39.

Pre-Upgrade Tasks

Before you upgrade to SAS Business Rules Manager 2.2, you should record the database settings in your current environment. You must enter this information in SAS Deployment Wizard.

- If you are using the SAS Web Infrastructure Platform Data Server for your SAS Business Rules Manager 2.1 database, record the database name and the user name for your database. The default database name is **brmdb**.

- If you are using Oracle for your database, record the following information:

  - fully qualified host name of the database server

  - port number of the database server

  - Oracle Site Identifier (SID)

  - user ID of the database user whose credentials are used to access SAS Business Rules Manager data on the server

  - password of the database user whose credentials are used to access SAS Business Rules Manager data on the server

  You can find the Oracle SID in the tnsnames.ora file. You can also determine the SID by running the following query using a database user ID on your Oracle instance:

  ```
  select instance from v$thread
  ```

  You must enter the SID in the **Database name** field in the SAS Decision Manager Database JDBC Properties dialog box in SAS Deployment Wizard.

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 for the SAS Decision Manager database. See Table 3.2 on page 18 for information about the tnsnames.ora file.

Post-Upgrade Steps for SAS Web Infrastructure Platform Data Server

You need to run PostgreSQL commands to upgrade the database. These commands are located in the following directory:

!SASHOME/SASWebInfrastructurePlatformDataServer/9.4/bin

Note: If you are performing the upgrade on the UNIX operating environment and these commands fail, you might need to set environment variables.See "Set Environment Variables for UNIX" on page 43 for more information.

When you run these commands, substitute the correct values for the host name, port number, and user ID for your database. These commands prompt you to enter a password.

1. On the upgraded server, run the **pg_dump** command. This command creates a script file that contains the SQL commands that are needed to reconstruct the database:

   ```
   pg_dump --host=host_name --port=nnnn --username=database_owner_userID
   --password -Fp -O brmdb > brmdb.dump
   ```

The –O option exports the database without commands that set ownership of any of the database objects.

2. Stop SASServer7.

3. If SAS Model Manager 13.1 is installed on the upgraded server, run the following command to create a backup of the SAS Model Manager tables:

   ```
   pg_dump --host=host_name --port=nnnn --username=userID
   --password -t sas mm* database_name > mmbackup-131.dump
   ```

   Note: Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

4. On the upgraded server, run the **dropdb** command to drop the database that was created by the SAS Deployment Wizard:

   ```
   dropdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

5. On the upgraded server, run the **createdb** command to create a new, empty database:

   ```
   createdb --host=host_name --port=nnnn --username=database_owner_userID
   --password database_name
   ```

6. On the upgraded server, restore the exported database:

   ```
   psql --host=host_name --port=nnnn -f brmdb.dump database_name userID
   ```

   Note: Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

   Note: You might see messages such as **no privileges were granted for "public"** or **role name does not exist**. These messages mean that these users are no longer applicable to the upgraded database instance and can be ignored.

7. Run the **migration_brm_2.1_to_brm_2.2_sql** migration script that is located in the following directory:

   **!SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/
   2.2/Config/Deployment/dbscript/postgres/migration**

   Note: In a multi-tier environment, copy the script to the server tier before you run it.

8. Run the **edm_workflow_interface_create_*.sql** scripts. These scripts are located in the following directory:

   **!SASHOME/SASDecisionManagerCommonMidTierforDecisionManager/
   2.2/Config/Deployment/dbscript/postgres**

   Note: In a multi-tier environment, copy the scripts to the server tier before you run them.

   Run the following scripts, in the order listed:

   1. edm_workflow_interface_create_table.sql

   2. edm_workflow_interface_create_sequence.sql

   3. edm_workflow_interface_create_constraint.sql

   4. edm_workflow_interface_create_trigger.sql

   5. edm_workflow_interface_required_inserts.sql

9. Restore the SAS Model Manager tables by using the backup that you created in Step 3:

    ```
    psql --host=host_name --port=nnnn -f mmbackup-131.dump database_name userID
    ```

    Note: Specify the user ID for Decision Manager Common Middle Tier. Do not specify the user ID of the database owner account.

10. 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 32 for more information.

11. Update your user group memberships as needed. See "Update User Group Membership" on page 42 for more information.

Chapter 8
Configuring Users, Groups, and Roles

Security Administration Tasks for SAS Business Rules Manager

Security administration for SAS Business Rules 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* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

Administering SAS Identities for Users

Overview of SAS Identities

For each SAS Business Rules 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.

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

Table 8.1 *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 Appendix 2, "User Import Macros," in *SAS Intelligence Platform: Security Administration Guide*. These documents are available at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

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 Business Rules Manager

Table 8.2 *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.
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 Decision Manager Common: Administration and Business Rules Manager: All Capabilities roles.

Group	Description
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.
	It is associated with the necessary identity in order to allow users to access the business rules database during rule flow execution. This group is the only group that is granted permission to publish business rules content to the SAS Content Server by default. Unless you make configuration changes, membership in this group is required for business rules users who do not have administrator permission.

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 Business Rules Manager

Your installation includes several predefined roles for administrators and users of SAS Business Rules 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 8.3 Predefined User Roles and Capabilities

Role	Description
Decision Manager Common: Administration	Users in this role can 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.

Role	Description
Business Rules Manager: All Capabilities	Users in this role can view and edit all business rules content, including vocabularies, entities, terms, lookup tables, rule sets, and rule flows.
Business Rules Manager: Rule Flow and Rule Set Designer	Users in this role can create, edit, and delete rule sets and rule flows.
Business Rules Manager: Rule Flow and Rule Set Read-Only	Users in this role can view rule sets and rule flows.
Business Rules Manager: Vocabulary and Lookup Designer	Users in this role can create, edit, and delete vocabularies, entities, terms, and lookup tables.
Business Rules Manager: Vocabulary and Lookup Read-Only	Users in this role can view vocabularies, entities, terms, and lookup tables.
Comments: Administrator	Users in this roll can 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.

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 Business Rules Manager" on page 52.
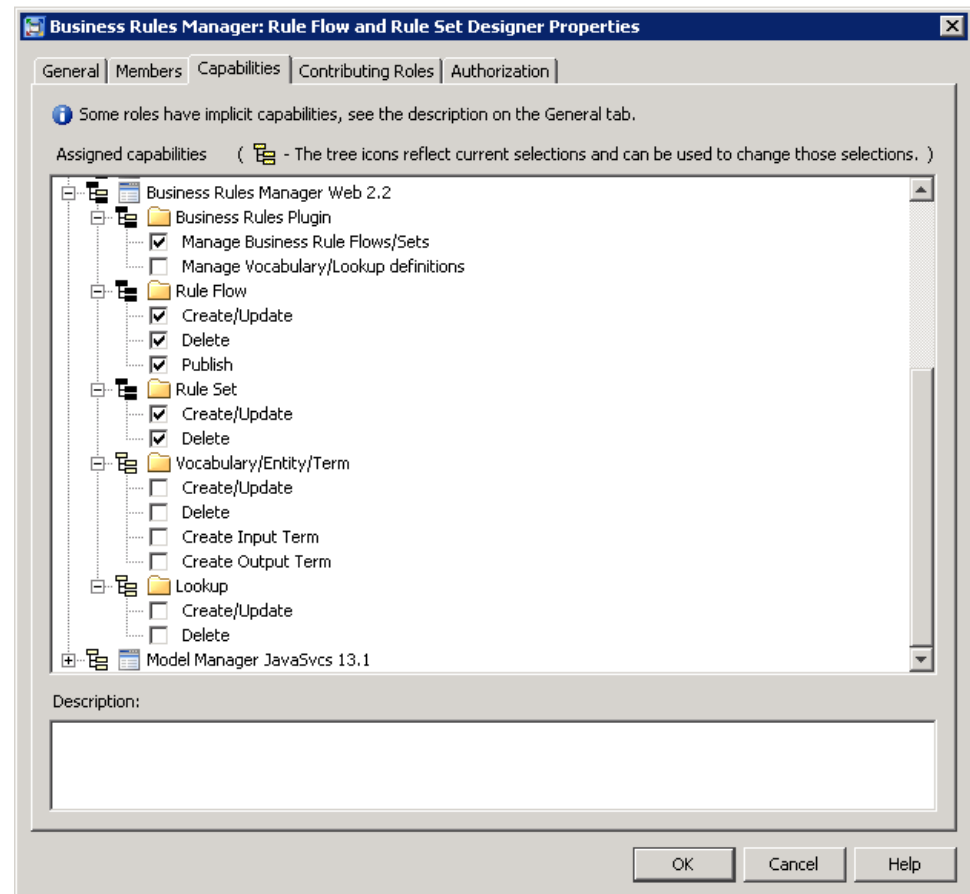
Note: Some roles have implicit capabilities that are not specified on the **Capabilities** tab.

The SAS Business Rules 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 Business Rules 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.

Icon	Description
	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.

Administering Group and Role Membership

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

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* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

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* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

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.

- Back up the metadata server before modifying roles, and 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* at http://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

Chapter 9
Configuring SAS Workflow

Configuring SAS Workflow for Use with SAS Business Rules 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 Business Rules Manager is used to manage the workflows that are associated with versions. For more information about SAS Workflow, see "SAS Workflow" in Chapter 1 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

To use SAS Workflow with SAS Business Rules 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.

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

Guidelines for Creating Workflow Definitions

When you create workflow definitions in SAS Workflow Studio to use with SAS Business Rules 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 Business Rules Manager.

- Only the Potential Owner and Business Administrator workflow roles are used by SAS Business Rules 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 Business Rules 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

  For more information, see "Security Administration Tasks for SAS Business Rules Manager" on page 49.

- Only workflow definitions that are activated in the Workflow repository, that are associated with the `mmapi` tag attribute in the file properties, are available toSAS Business Rules Manager.

How to Add the Approval Attribute to a Status

The `Approval` attribute allows a workflow designer to signify that a specific task approves the associated version for a rule flow. This attribute then notifies the users of the version that a rule flow is approved. For business rules the `Approval` attribute must be set so that a workflow can be used to manage rule flows.

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**.
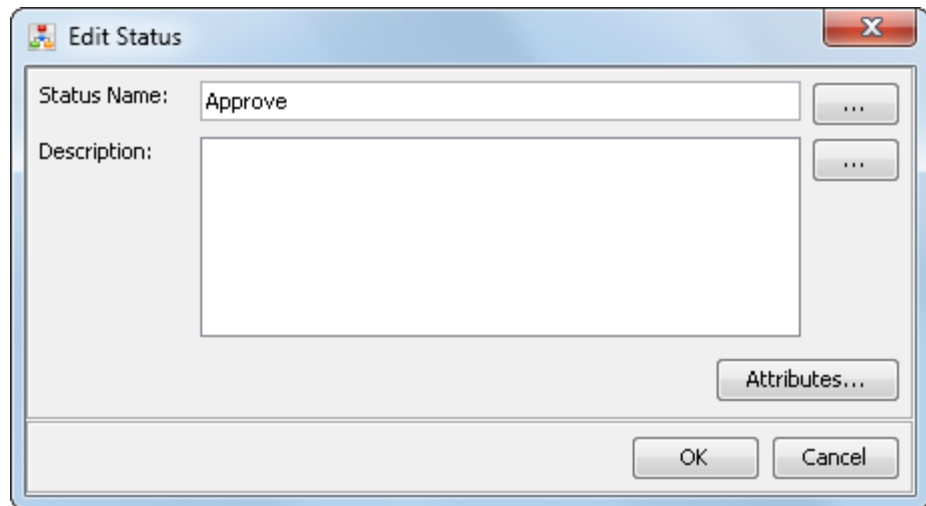
3. Click **Attributes**.

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 Business Rules Manager

After you have created a workflow definition in the SAS Workflow Studio, you must make the workflow definition available to SAS Business Rules 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 Chapter 5 of *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 Appendix 1, "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 Business Rules 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 Business Rules Manager

To verify that the workflow definitions are available in the Workflows category view of SAS Business Rules 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 Chapter 10 of *SAS Business Rules Manager: User's Guide*.

Configuring 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 Chapter 7 of *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*.

Part 3

Import and Export Macros

Import And Export Macro Reference

Introduction to the Import and Export Macros

SAS Business Rules Manager provides the following macros for importing data into the rules database and exporting data from the rules 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 rules database into a CSV file

%BRM_EXPORT_RULESET
    exports rule sets from the rules database into a CSV file

%BRM_EXPORT_VOCABULARY
    exports vocabularies from the rules database into a CSV file

%BRM_IMPORT_FOLDER
>   imports the folder definitions that are in the specified CSV file into the rules database.

%BRM_IMPORT_LOOKUP
>   imports lookup tables from the specified CSV file into the rules database.

%BRM_IMPORT_RULE_FLOW
>   imports rule flows from a CSV file into the rules database

%BRM_IMPORT_RULESET
>   imports rule sets from a CSV file into the rules database

%BRM_IMPORT_VOCABULARY
>   imports vocabulary terms from a CSV file into the rules database

%BRM_LOAD_VOCABULARY
>   loads the vocabulary terms into the WORK.TERM data set that was created by the %BRM_CREATE_TEMP_TERM macro.

Dictionary

%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.

> **Restriction:** This macro must be run on the server tier.

Syntax

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

Required Argument

DATAFILE=*input_file*
>   specifies either a SAS data set name or the full pathname to a CSV file. 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 rules database and is displayed in the SAS Business Rules Manager interface.

Default   User ID of the user that is running the macro

Details

This macro reads a CSV file or SAS data set that defines vocabulary terms and that 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 rules database. See "%BRM_LOAD_VOCABULARY" on page 82 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 10.1.

Table 10.1   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.

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

Syntax

%BRM_EXPORT_FOLDER (CSV=*output_filename*.CSV<, FOLDER_PATH=*path_name*>);

Required Argument

CSV=*output_filename*
>   specifies the full pathname to the CSV file where you want to export the data.

Optional Argument

FOLDER_PATH=*<path_name>*
>   specifies a business rules folder that you want to export. By default,
>   %BRM_EXPORT_FOLDER exports all folders. You do not need to specify the
>   FOLDER_PATH= option unless you want to export a specific folder.

>   **Example**   FOLDER_PATH=Loans/Retail/Applications

%BRM_EXPORT_LOOKUP

Exports the contents of lookup tables into a CSV file.

>   **Restriction:**   This macro must be run on the server tier.

Syntax

%BRM_EXPORT_LOOKUP (CSV=*output_filename*.CSV<, FOLDER_PATH=*path-name*>
<, LOOKUP='*lookup_table_1*'<, '*lookup_table_2*'>...>);

Required Argument

CSV=*output_filename*
>   specifies the full pathname to the CSV file where you want to export the data.

Optional Arguments

FOLDER_PATH=*path_name*
>   specifies a business rules folder pathname in SAS Business Rules Manager that you
>   want to filter the output by. If you specify a folder pathname, then only the objects in
>   that path are exported.

>   Use a forward slash to separate folder names.

>   **Example**   FOLDER_PATH=Loans/Retail/Applications

LOOKUP=%STR('*lookup_table_1*'<, '*lookup_table_2*'>...)
>   specifies the lookup tables that you want to export. Specify the names of the lookup
>   tables, enclosed 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.

>   **Tip**   You can filter the lookup tables that are exported by specifying the
>   FOLDER_PATH= option.

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

%BRM_EXPORT_RULE_FLOW

Exports rule flows into a CSV file.

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

Syntax

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

Required Arguments

CSV=*output_filename*
specifies the full pathname to the CSV file where you want to export the 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 enclosed in quotation marks. Separate multiple identification numbers with commas.

Tip You can filter the rule flows that are exported by specifying the FOLDER_PATH= option.

Example `ruleflows=%str(10168,10043)`

Optional Argument

FOLDER_PATH=*path_name*
specifies a business rules folder pathname in SAS Business Rules Manager that you want to filter the output by. If you specify a folder pathname, then only the objects in that path are exported. For example, if you specify RULEFLOWS=ALL and FOLDER_PATH=RetailLoans, then only the rule flows in the folder RetailLoans are exported. If you specify RULEFLOWS=%STR(10045,10572) and FOLDER_PATH=RetailLoans, but neither of the specified rule flows are in the RetailLoans folder, then no rule flows are exported.

Use a forward slash to separate folder names.

Example FOLDER_PATH=Loans/Retail/Applications

%BRM_EXPORT_RULESET

Exports rule sets from the rules database into a CSV file.

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

Syntax

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

Required Arguments

CSV=*output_filename*
> specifies the full pathname to the CSV file where you want to export the 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 enclosed in quotation marks. Separate multiple identification numbers with commas.

> | Tip | You can filter the rule sets that are exported by specifying the FOLDER_PATH= option. |
> |---|---|

> | Example | `rulesets=%str(168,43)` |
> |---|---|

Optional Argument

FOLDER_PATH=*path_name*
> specifies a business rules folder pathname in SAS Business Rules Manager that you want to filter the output by. If you specify a folder pathname, then only the objects in that path are exported. For example, if you specify RULESETS=ALL and FOLDER_PATH=RetailLoans, then only the rule sets in the folder RetailLoans are exported. If you specify RULESETS=%STR(10045,10572) and FOLDER_PATH=RetailLoans, but neither of the specified rule sets are in the RetailLoans folder, then no rule sets are exported.

> Use a forward slash to separate folder names.

> | Example | FOLDER_PATH=Loans/Retail/Applications |
> |---|---|

%BRM_EXPORT_VOCABULARY

Exports vocabularies from the rules database into a CSV file.

> | Restriction: | This macro must be run on the server tier. |
> |---|---|

Syntax

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

Required Arguments

CSV=*output_filename*
> specifies the full pathname to the CSV file where you want to export the data.

VOCAB=ALL | %STR('*vocabulary_1*'<, '*vocabulary_2*'>...)
> specifies 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 identification numbers with commas.

Tip   You can filter the vocabularies that are exported by specifying the FOLDER_PATH= option.

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

Optional Argument

FOLDER_PATH=*path_name*
specifies a business rules folder pathname in SAS Business Rules Manager that you want to filter the output by. If you specify a folder pathname, then only the objects in that path are exported. For example, if you specify VOCAB=ALL and FOLDER_PATH=RetailLoans, then only the vocabularies in the folder RetailLoans are exported. If you specify VOCAB=%STR('loanVocab','riskVocabulary') and FOLDER_PATH=RetailLoans, but neither of the specified vocabularies are in the RetailLoans folder, then no vocabularies are exported.

Use a forward slash to separate folder names.

Example   FOLDER_PATH=Loans/Retail/Applications

%BRM_IMPORT_FOLDER

Imports the folder definitions that are in the specified CSV file into the rules database.

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

The same user can run any of import macros at the same time. However, different users cannot run the same import macro simultaneously.

Syntax

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

Required Arguments

CSV=*input_filename*
specifies the full pathname to the CSV file where you want to import the data from.

REJECT=*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 rules database. See "Using the %BRM_IMPORT_FOLDER Macro" on page 72 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 rules database and is displayed in the SAS Business Rules Manager interface.

Default   User ID of the user that is 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 path 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 import data to a folder named Applications and to specify a blank column for the folder description and default folder flag, specify the following in the CSV file:

```
Applications,,Y,,Loans/Retail
```

Table 10.2 *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
DEFAULT_FOLDER_FLG	Specifies whether the folder is the default folder. Specify **Y** or **N**.	Yes
FOLDER_PATH	The pathname to the business rules folder where you want to import the contents of the CSV file. This path must exist. Separate folder names with forward slashes.	No

%BRM_IMPORT_LOOKUP

Imports lookup tables from the specified CSV file into the rules database.

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

The same user can run any of import macros at the same time. However, different users cannot run the same import macro simultaneously.

Syntax

%BRM_IMPORT_LOOKUP (CSV=*input_filename*.CSV,
REJECT=*reject_filename*.CSV*<, options>*);

Required Arguments

CSV=*input_filename*
    specifies the full pathname to the CSV file where you want to import the data from.

REJECT=*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 rules database. See "Using the %BRM_IMPORT_LOOKUP Macro" on page 73 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 rules database and is displayed in the SAS Business Rules Manager interface.

    Default   User ID of the user that is running the macro

BYPASSLOCK=Y|N
    enables you to override the lock that another user has on the importing process. See "Using the %BRM_IMPORT_LOOKUP Macro" on page 73 for more information.

    Default   N

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 path to determine whether a lookup table already exists. If the lookup table already exists, then it is updated. If the path exists but the lookup table does not exist, the lookup table is created. If the path 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.

When you run the %BRM_IMPORT_LOOKUP macro, it creates a lock table in the rules database named lock_import_lookup table. The SAS log states which user holds the lock and the time at which the lock started. This lock might remain in place after the macro has finished. If this happens, you can override the lock by specifying the BYPASSLOCK=Y option when you run the macro.

Format of the Lookup CSV Input File

Each 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. For example, to import the key **AU** and the value **Australia** into the lookup table **Country_Codes** and to specify a blank column for the description, specify the following in the CSV file:

```
Country_Codes,,AU,Australia,Loans/Retail
```

Table 10.3 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 path 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 rules database.

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

The same user can run any of import macros at the same time. However, multiple users cannot run the same import macro simultaneously.

Syntax

%BRM_IMPORT_RULE_FLOW(CSV=*input_filename*.CSV, REJECT=*reject_filename*.CSV*<, options>*);

Required Arguments

CSV=*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 75.

REJECT=*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 rules database. See "Using the %BRM_IMPORT_RULE_FLOW Macro" on page 75 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 rules database and is displayed in the SAS Business Rules Manager interface.

> **Default** User ID of the user that is running the macro

BYPASSLOCK=Y|N

> enables you to override the lock that another user has on the importing process. See for more information.

> **Default** N

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 rule flow path to determine whether a rule flow already exists. If the rule flow path and name already exist, then the rule flow is updated. If the rule flow path exists but the rule flow name does not exist, the rule flow is created. If the rule flow path 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.

When you run the %BRM_IMPORT_RULE_FLOW macro, it creates a lock table in the rules database named lock_import_rule_flow. The SAS log states which user holds the lock and the time at which the lock started. This lock might remain in place after the macro has finished. If this happens, you can override the lock by specifying the BYPASSLOCK=Y option when you run the macro.

Format of the Rule Flow CSV Input File

Each row of the CSV input file identifies a rule set and a rule flow, and each row 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 the main section of the rule flow named assignRisk and to specify a blank column for the rule flow description, specify the following in the CSV file:

```
assignRisk,,main
```

Table 10.4   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 Chapter 8 of *SAS Business Rules 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 path 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 to 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
BY_TERM	The list of BY-group terms that the rule set uses. Separate multiple BY-group terms with commas. 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 Chapter 8 of *SAS Business Rules Manager: User's Guide* for more information.	Yes

Column	Description	Can Column Be Blank
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 rules database.

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

The same user can run any of import macros at the same time. However, multiple users cannot run the same import macro simultaneously.

Syntax

%BRM_IMPORT_RULESET (CSV=*input_filename*.CSV, REJECT=*reject_filename*.CSV*<, options>*);

Required Arguments

CSV=*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 78.

REJECT=*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 rules database. See "Using the %BRM_IMPORT_RULESET Macro" on page 78 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 rules database and is displayed in the SAS Business Rules Manager interface.

Default User ID of the user that is running the macro

BYPASSLOCK=Y|N
enables you to override the lock that another user has on the importing process. See "Using the %BRM_IMPORT_RULESET Macro" on page 78 for more information.

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 path to determine whether a rule set already exists. If the rule set path and name already exist, then the rule set is updated. If the rule set path exists but the rule set name does not exist, the rule set is created. If the rule set path 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.

When you run the %BRM_IMPORT_RULESET macro, it creates a lock table in the rules database named lock_import_rule_set. The SAS log states which user holds the lock and the time at which the lock started. This lock might remain in place after the macro has finished. If this happens, you can override the lock by specifying the BYPASSLOCK=Y option when you run the macro.

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, to add a rule to the rule set named assignRisk that uses the loanVocab vocabulary and to specify a blank column for the rule set description, specify the following in the CSV file:

```
assignRisk,,loanVocab
```

Table 10.5   *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 path must exist. Separate folder names with forward slashes.	No

Column	Description	Can Column Be Blank
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 Chapter 7 of *SAS Business Rules 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 terms that SAS Business Rules Manager displays at the top or left side of the decision table. These terms appear in the column headings of the decision table when you are viewing the decision table in the horizontal format. They appear in the row headings of the decision table when you are viewing the decision table in the vertical format.	No
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 decision table. See "Defining New Rules in the Rule Set" in Chapter 7 of *SAS Business Rules 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 rules database.

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

The same user can run any of import macros at the same time. However, the same import macro cannot be run simultaneously by different users.

Syntax

%BRM_IMPORT_VOCABULARY (CSV=*input_filename*.CSV,
REJECT=*reject_filename*.CSV*<, options>*);

Required Arguments

CSV=*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 Vocabulary CSV Input File" on page 81.

REJECT=*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 rules database. See "Using the %BRM_IMPORT_VOCABULARY Macro" on page 80 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 rules database and is displayed in the SAS Business Rules Manager interface.

  Default User ID of the user that is running the macro

BYPASSLOCK=Y|N
  enables you to override the lock that another user has on the importing process. See "Using the %BRM_IMPORT_VOCABULARY Macro" on page 80 for more information.

  Default N

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.

When you run the %BRM_IMPORT_VOCABULARY macro, it creates a lock table in the rules database named lock_import_vocabulary. The SAS log states which user holds the lock and the time at which the lock started. It is possible for this lock to remain in place after the macro has finished. If this happens, you can override the lock by specifying the BYPASSLOCK=Y option when you run the macro.

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, to add a term to the entity named Customer in the vocabulary named loanVocab and to specify a blank column for the vocabulary description, specify the following in the CSV file:

```
loanVocab,,Customer
```

Table 10.6   *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
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 Chapter 5 of *SAS Business Rules Manager: User's Guide* for more information.	Yes

Column	Description	Can Column Be Blank
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 path must exist. Separate folder names with forward slashes.	No

%BRM_LOAD_VOCABULARY

Loads the vocabulary terms in the WORK.TERM data set that was created by the %BRM_CREATE_TEMP_TERM macro.

Syntax

%BRM_LOAD_VOCABULARY (FOLDER_PATH=*path*,
VOCAB_NM=*vocabulary-name*,
VOCAB_ENTITY_NM=*entity-name<, options>*);

Required Arguments

FOLDER_PATH=*path-name*
 specifies the pathname to the business rules folder where you want to import the vocabulary terms. Separate folder names with forward slashes.

Requirement	The path must exist. If the path does not exist, the macro terminates and writes an error message to the SAS log.
Example	FOLDER_PATH=Loans/Retail/Applications

VOCAB_NM=*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=*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 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 rules database and is displayed in the SAS Business Rules Manager interface.

Default   User ID of the user that is running the macro

BYPASSLOCK=Y|N

enables you to override the lock that another user has on the importing process.

Default   N

Details

When you run the %BRM_LOAD_VOCABULARY macro, it creates a lock table in the rules database named lock_import_vocabulary. The SAS log states which user holds the lock and the time at which the lock started. It is possible for this lock to remain in place after the macro has finished. If this happens, you can override the lock by specifying the BYPASSLOCK=Y option when you run the macro.

Index

Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.