



SAS[®] 9.1.3 Intelligence Platform

Desktop Application Administration Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2006. *SAS® 9.1.3 Intelligence Platform: Desktop Application Administration Guide*. Cary, NC: SAS Institute Inc.

SAS® 9.1.3 Intelligence Platform: Desktop Application Administration Guide

Copyright © 2002-2006, SAS Institute Inc., Cary, NC, USA

ISBN-13: 978-1-59994-287-2

ISBN-10: 1-59994-287-9

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

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

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

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

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

1st printing, November 2006

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

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Contents

What's New v

Overview v

Documentation Enhancements v

Chapter 1 △ **Introduction to Desktop Application Administration** 1

About the Desktop Applications 1

Installation Notes for Desktop Applications 2

Accessibility Features in the SAS Desktop Clients 2

Chapter 2 △ **Administering the SAS Add-In for Microsoft Office** 3

Overview of Administering the SAS Add-In for Microsoft Office 3

Connecting the SAS Add-In for Microsoft Office into the SAS Intelligence Platform 3

Reviewing Library Access for the SAS Add-In for Microsoft Office 4

Chapter 3 △ **Administering SAS Data Integration Studio** 5

Overview of Administering SAS Data Integration Studio 6

Connecting to SAS Servers 8

Connecting to Data Servers 10

Setting Up Change Management 11

Setting the Default umask on UNIX Systems 16

Setting Up Multi-Tier Environments 16

Using Custom Tree Folders for Security 25

Importing and Exporting Generated Transformations 28

Importing and Exporting Metadata 29

Administering Message Queues 31

Providing Support for Web Services 34

Enabling the Bulk Loading of Data 36

Testing the Platform Computing Scheduling Server 37

Setting Up the SAS Data Quality Server Software 39

Redirecting Output and Logging Information to a File 43

Enabling Status Code Handling 45

Enabling the External File Wizards to Retrieve Files Using FTP or HTTP 46

Chapter 4 △ **Administering SAS Enterprise Guide** 49

Overview of Administering SAS Enterprise Guide 49

Connecting SAS Enterprise Guide to SAS Servers in the SAS Intelligence Platform 49

Reviewing Library Access in SAS Enterprise Guide 50

Enabling ESRI Maps in SAS Enterprise Guide 52

Chapter 5 △ **Administering SAS Enterprise Miner** 53

Overview of Preparing SAS Enterprise Miner for Use 53

Configuring SAS Enterprise Miner 54

Customizing SAS Enterprise Miner Metadata Using the Enterprise Miner Plug-in for SAS Management Console	57
The Projects Folder	58
Customize Servers for Enterprise Miner	59
The Models Folder	60
Customizing the Apache Tomcat HTTP Server	61
Securing SAS Enterprise Miner Metadata	63
Chapter 6 △ Administering SAS Information Map Studio	67
Overview of Administering SAS Information Map Studio	67
Managing Storage for Information Maps	68
Adding Content for Use by SAS Information Map Creators	68
Considerations for Setting Up Access Control for Information Maps	69
Setting Up Users for SAS Information Map Studio	70
Understanding the Configuration of SAS Information Map Studio	70
Managing the Log File in SAS Information Map Studio	71
Managing the Size of the Java Heap	71
Chapter 7 △ Administering SAS Management Console	73
SAS Management Console Overview for Administrators	73
Quick Reference for SAS Management Console	73
Understanding the Configuration of SAS Management Console	75
Chapter 8 △ Administering SAS OLAP Cube Studio	77
SAS OLAP Cube Studio Overview for Administrators	77
Setting Up the SAS Intelligence Platform for SAS OLAP Cube Studio	78
Understanding the Configuration of SAS OLAP Cube Studio	78
Appendix 1 △ Recommended Reading	79
Recommended Reading	79
Glossary	81
Index	83

What's New

Overview

The desktop applications of the SAS Intelligence Platform provide a comprehensive suite of knowledge generation and reporting capabilities in a series of graphical interfaces. The new administrator's guide, *SAS Intelligence Platform: Desktop Application Administration Guide*, shows you how to configure and manage the desktop application across your enterprise.

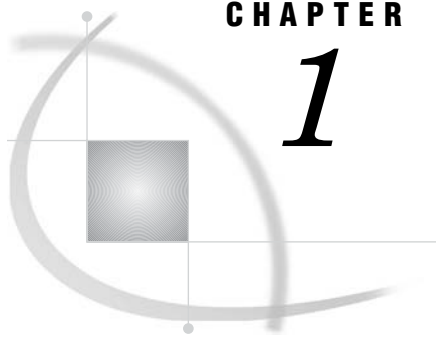
Documentation Enhancements

The *SAS Intelligence Platform: Desktop Application Administration Guide* addresses all of the administrative tasks for the following desktop applications:

- SAS Add-In for Microsoft Office
- SAS Data Integration Studio
- SAS Enterprise Guide
- SAS Enterprise Miner
- SAS Information Map Studio
- SAS Management Console
- SAS OLAP Cube Studio

For some of the desktop applications, the information in the new document complements existing information in other documents. For example, it documents administrative tasks for SAS Enterprise Guide that are specific to the SAS Intelligence Platform. Other administrative tasks are documented in *Administering SAS Enterprise Guide*.

Most of the content in the new document was formerly part of the *SAS Intelligence Platform: Administration Guide*. For information about the administrative documentation for the SAS Intelligence Platform, see <http://support.sas.com/documentation/configuration/913admin.html>.



CHAPTER

1

Introduction to Desktop Application Administration

About the Desktop Applications 1
Installation Notes for Desktop Applications 2
Accessibility Features in the SAS Desktop Clients 2

About the Desktop Applications

The SAS Intelligence Platform includes the following client applications:

SAS Add-In for Microsoft Office

brings the power of SAS into the toolbar of Microsoft Excel, Word, and PowerPoint.

SAS Data Integration Studio

enables enterprise data collection, analysis, reporting, and publishing using a graphical process designer and numerous data transformation templates.

SAS Enterprise Guide

enables the creation of stored processes and the querying of SAS OLAP cubes.

SAS Enterprise Miner

provides a point-and-click Java interface for creating predictive models that identify trends, opportunities, and threats based on the analysis of data from across your enterprise.

SAS Information Map Studio

defines data collections from across your enterprise for use in analysis and reporting.

SAS Management Console

provides the primary administrative interface for the SAS Intelligence Platform.

SAS OLAP Cube Studio

provides a straightforward wizard format for building and tuning SAS OLAP cubes.

Administrative information for client applications that run in a Web browser is provided in the *SAS Intelligence Platform: Web Application Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>. The Web applications include SAS Web Report Studio, SAS Web OLAP Viewer, and the SAS Information Delivery Portal.

Administrative tasks for the desktop applications consist of installation, configuration, and fault diagnosis using log files. Specific administrative tasks differ between clients.

Installation Notes for Desktop Applications

You initially install desktop applications as specified in your deployment plan file, using the SAS Software Navigator and the SAS Configuration Wizard. After you complete the initial deployment, you can install additional clients by using a software index installation. These subjects are addressed in the *SAS Intelligence Platform: Installation Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

Regardless of when you install a desktop application, it is important for you to know that client configuration may not end with the SAS Configuration Wizard. Check the chapter in this document for each client to ensure that you complete the configuration process. You may need to configure libraries, update user permissions, or change memory allocations.

Accessibility Features in the SAS Desktop Clients

For information about accessibility for any of the products mentioned in this book, see the documentation for that product.

For Windows-based assistive technologies to interoperate with SAS Java-based desktop applications, you need to download the Java Access Bridge (JAB) from Sun Microsystems at this URL: <http://www.sas.com/govedu/java.html>. The SAS Java-based applications that run on Windows are listed under “Clients” in the “Architecture of the SAS Intelligence Platform” chapter of the *SAS Intelligence Platform: Overview*.

After downloading the JAB, you need to customize the standard installation procedure to facilitate access to it. This setup assumes that Java is already installed on the machine. A known risk is that SAS applications can be corrupted by JAR files that might be installed in the `jre/lib/ext` directory. To solve this problem, SAS has implemented a custom class loader to replace the system class loader. The custom class loader loads only approved extensions that SAS configures based on the JRE the vendor provides. A consequence of hiding the existing directory is that it disables installed accessibility-related applications. To configure SAS Java applications to permit access to the Java Access Bridge:

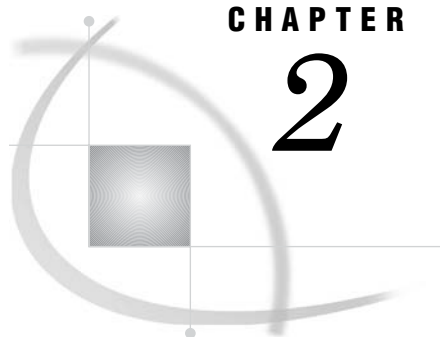
- 1 On the client machine, locate all copies of the file `sas.java.ext.config`.
- 2 In each instance of the file, edit the file in a plain text editor, such as Microsoft Notepad.
- 3 Add the following two lines to the bottom of the file:

```
access-bridge.jar
jaccess-1_4.jar
```

Note: The filename of the `jaccess*.jar` file depends on the version of the JRE. Listed here is the filename for Version 1.4. If you are using a different version, your filename will vary.

- 4 Save the changes and close the file.

If you have questions or concerns about the accessibility of SAS products, send e-mail to accessibility@sas.com.



CHAPTER

2

Administering the SAS Add-In for Microsoft Office

<i>Overview of Administering the SAS Add-In for Microsoft Office</i>	3
<i>Connecting the SAS Add-In for Microsoft Office into the SAS Intelligence Platform</i>	3
<i>Reviewing Library Access for the SAS Add-In for Microsoft Office</i>	4

Overview of Administering the SAS Add-In for Microsoft Office

The SAS Add-In for Microsoft Office is a Component Object Model (COM) add-in that builds the power of SAS data access, analysis, and reporting directly into the user interface of Microsoft Excel, Word, and PowerPoint. To see the SAS functionality in these products, open the **SAS** menu in the toolbar.

To learn more about the SAS Add-In for Microsoft Office, select **SAS ► Help ► SAS Add-In for Microsoft Office Help** in Excel, Word, or PowerPoint. Additional information is available at the product home page, at http://www.sas.com/technologies/bi/query_reporting/addin/.

Post-installation administrative tasks for the SAS Add-In for Microsoft Office consist of defining a connection to the SAS Metadata Server and reviewing libraries to ensure appropriate access.

Connecting the SAS Add-In for Microsoft Office into the SAS Intelligence Platform

Follow these steps to define or change a SAS server connection in Excel, Word, or PowerPoint:

- 1 Start Excel, Word, or PowerPoint.
- 2 Select **SAS ► Tools ► Server Connections**.
- 3 In the Server Connections window, click **Change**.
- 4 In the Connect to SAS Metadata Server window, select **Connect using the following parameters**.
- 5 In the **Machine Name** field, enter a non-relative name of the SAS server:

dd958.na.ourco.com

To access libraries of tables, connect to the host that runs the SAS Metadata Server. To access SAS OLAP cubes, connect to the host that runs the SAS OLAP Server that hosts those cubes.

- 6 Accept the default values for **Protocol** and **Port**.

- 7 In the **Authentication Domain** field, enter the name of the domain that applies to that server. In most cases you will simply accept the default value, **DefaultAuth**.
- 8 Have the user enter the user name and password that apply to the server's authentication domain. In most cases, authentication is provided by the host operating system.
- 9 Click **OK** to connect to the SAS server.
- 10 In the **Select repository to use** field, select a metadata repository. A typical selection is **Foundation**.
- 11 If you want the server connection information to appear by default when you create repositories in other Windows applications on the same host (Excel, Word, and PowerPoint), select **Save as the default for all SAS applications**.
- 12 If you want the user name and password to be provided by default when you create server connections in other applications on the same host, select **Save user name and password with user information**.
- 13 Select a default SAS server for this server connection. A typical selection is **SASMain**.
- 14 At this point you should be connected to the SAS server and the **Connect** button should be unavailable. Click **OK** to close the window.

For more information on authentication, see “Making Preliminary Decisions about Authentication” in the *SAS Intelligence Platform: Security Administration Guide*.

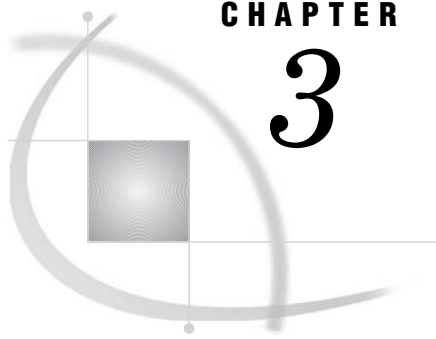
Reviewing Library Access for the SAS Add-In for Microsoft Office

When the SAS Add-In for Microsoft Office writes data to a table that is registered on the SAS Metadata Server, and when that update changes the structure of the table, the metadata for that table is not updated on the SAS Metadata Server. For this reason, you need to regulate Write access to registered tables from Excel, Word, and PowerPoint. Read access, for the purposes of querying SAS OLAP cubes using Excel for example, requires no special consideration on your part.

Another access issue that you might need to consider regards the fact that the SAS Add-In for Microsoft Office does not enforce Read, Write, Create, or Delete permissions on tables or columns, as defined on the SAS Metadata Server. In general, Add-In clients should be denied access to libraries that implement metadata access control.

Access to registered libraries is determined in part by the library access method. By default, all libraries that are registered on the SAS Metadata Server are read-only. Add-In users cannot create, add, or delete tables, and metadata access control is ignored. The default library access method can be changed by using SAS Enterprise Guide, as described in “Reviewing Library Access in SAS Enterprise Guide” on page 50.

To maintain referential integrity on the SAS Metadata Server, you may need to inform your Add-In users about the nature of their permissions to access registered libraries.



CHAPTER

3

Administering SAS Data Integration Studio

<i>Overview of Administering SAS Data Integration Studio</i>	6
<i>Connecting to SAS Servers</i>	8
<i>Overview of SAS Server Connections</i>	8
<i>Connecting to a Metadata Server</i>	8
<i>Connecting to a Workspace Server</i>	9
<i>Connecting to Data Servers</i>	10
<i>Setting Up Change Management</i>	11
<i>Overview of Setting Up Change Management</i>	11
<i>Creating a Repository Directory</i>	11
<i>Creating a Project Repository</i>	12
<i>Setting Metadata Permissions for Your User</i>	13
<i>Creating a Metadata Profile</i>	14
<i>Using the Metadata Profile</i>	15
<i>Responding When No Project Repositories Are Displayed</i>	15
<i>Setting the Default umask on UNIX Systems</i>	16
<i>Setting Up Multi-Tier Environments</i>	16
<i>Accessing Data Remotely</i>	16
<i>Overview of Remote Data Access</i>	16
<i>Setting Up a Local Host</i>	17
<i>Setting Up a Remote Host</i>	17
<i>Setting Up a Remote SAS/CONNECT Server</i>	18
<i>Defining a SAS Library</i>	18
<i>Setting Up a SAS Data Integration Studio Client Machine</i>	19
<i>Defining Tables in the Metadata for the Remote Host</i>	19
<i>Testing Your Setup Using View Data</i>	20
<i>Processing Jobs Remotely</i>	20
<i>Overview of Processing Jobs Remotely</i>	20
<i>Starting the SAS Management Console</i>	21
<i>Defining Your Application Server and Workspace Server</i>	21
<i>Editing the SAS Command for the Workspace Server</i>	22
<i>Defining Your SAS Object Spawner</i>	22
<i>Defining Your SAS Data Step Batch Server</i>	23
<i>Defining Your Batch Job Deployment Directory</i>	23
<i>Verifying Your Object Spawner</i>	24
<i>Verifying That the Object Spawner Is Working Properly</i>	24
<i>Deploying a Job Remotely Using SAS Data Integration Studio</i>	24
<i>Using Custom Tree Folders for Security</i>	25
<i>Overview of Custom Tree Folders</i>	25
<i>Setting Permissions on Custom Tree Folders</i>	25
<i>Administering Multiple Inheritance of Access Controls</i>	28
<i>Importing and Exporting Generated Transformations</i>	28

<i>Importing and Exporting Metadata</i>	29
<i>Requirements for Supporting Metadata</i>	29
<i>Managing the Supported Metadata Formats</i>	29
<i>Importing Metadata</i>	29
<i>Exporting Metadata</i>	30
<i>Administering Message Queues</i>	31
<i>What Are Message Queues?</i>	31
<i>About the Third-Party Messaging Software</i>	31
<i>About Message Queues in the SAS Intelligence Platform</i>	31
<i>Requirements for Message Queues</i>	31
<i>Installing and Configuring the Microsoft MSMQ Message Queue Software</i>	32
<i>Installing and Configuring the WebSphere MQ Message Queue Software</i>	32
<i>Registering Metadata for Queues</i>	33
<i>Managing Queues and Queue Managers</i>	33
<i>Providing Support for Web Services</i>	34
<i>Web Services</i>	34
<i>SAS BI Web Services</i>	34
<i>SAS BI Web Services for Java Versus SAS BI Web Services for .NET</i>	34
<i>Install SAS BI Web Services for .NET</i>	34
<i>Install SAS BI Web Services for Java</i>	35
<i>Calling Web Services</i>	36
<i>Enabling the Bulk Loading of Data</i>	36
<i>Bulk Loading Data into DBMS Tables</i>	36
<i>Enable the Workspace Server to Execute Operating System Commands</i>	37
<i>Testing the Platform Computing Scheduling Server</i>	37
<i>Setting Up the SAS Data Quality Server Software</i>	39
<i>Overview of SAS Data Quality Server Software</i>	39
<i>Understanding the Data Quality Configuration</i>	39
<i>Testing the SAS Data Quality Server Software</i>	40
<i>Downloading Locales</i>	41
<i>Creating Schemes</i>	42
<i>Setting Data Quality Options for SAS Data Integration Studio</i>	42
<i>Redirecting Output and Logging Information to a File</i>	43
<i>Enabling Status Code Handling</i>	45
<i>Overview of Status Code Handling</i>	45
<i>Supporting the E-mail Action</i>	45
<i>Supporting the Custom Action</i>	45
<i>Enabling the External File Wizards to Retrieve Files Using FTP or HTTP</i>	46

Overview of Administering SAS Data Integration Studio

SAS Data Integration Studio is a visual design tool that enables data integration specialists to consolidate and manage enterprise data from a variety of source systems, applications, and technologies. This software enables these users to create process flows that accomplish the following tasks:

- extract, transform, and load (ETL) data for use in data warehouses and data marts
- cleanse, migrate, synchronize, replicate, and promote data for applications and business services.

SAS Data Integration Studio enables users to integrate information from any platform that is accessible to SAS and from any format that is accessible to SAS.

Note: SAS Data Integration Studio was formerly named SAS ETL Studio. Δ

For information on how to build warehouses and marts, data integration specialists should consult the *SAS Data Integration Studio: User's Guide*, which is available at support.sas.com. To access the book, go to <http://support.sas.com>. Then select **Documentation ► Products & Solutions**. In the **Select a Product** drop-down list on the Documentation for SAS Products and Solutions page, select **SAS Data Integration Studio** and click **Go**.

There are also a number of administrative tasks that you must perform in support of these users. For example, in addition to installing the product, you might need to perform the following tasks:

Table 3.1 Administrative Tasks for SAS Data Integration Studio

Administrative Task	Purpose of Task
Connecting to SAS Servers	Make sure that your data integration specialists can connect to the necessary SAS servers. For instance, each user must be able to connect to the SAS Metadata Server to register data sources and other objects.
Connecting to Data Servers	Make sure that your data integration specialists can connect to the necessary data servers.
Setting Up Change Management	Set up a change-management system, which enables individual users to check objects out of a foundation repository and place them in a private repository (called a <i>project repository</i>) where the users can test changes.
Setting Up Multi-Tier Environments	Set up SAS Data Integration Studio to work in a multi-tier configuration. Almost all configurations have multiple tiers.
Using Custom-Tree Folders for Security	Set up a folder structure for metadata objects that enables you to control access to those objects.
Importing and Exporting Generated Transformations	Control access to SAS Data Integration Studio generated transformations, which are a class of user-written transformation.
Importing and Exporting Metadata	Import metadata from data modeling tools such as the AllFusion ERwin Data Modeler.
Administering Message Queues	Install third-party message queue software, define queues, and register queues for interprocess communication.
Providing Support for Web Services	Install SAS BI Web Services for Java or .NET so that Web services can invoke store processes.
Enabling the Bulk Loading of Data	Enable SAS Workspace Servers to execute external commands so that jobs can rapidly load data into a DBMS.
Testing the Platform Computing Scheduling Server	Test the servers (and clients) that enable your users to schedule sets of SAS Data Integration Studio jobs.
Setting Up a SAS Data Quality Server	Set up the infrastructure necessary for your users to employ data-quality transformations.

Administrative Task	Purpose of Task
Redirecting Output and Logging Information to a File	Specify alternative destinations for the SAS log and SAS output.
Enabling Status Code Handling	Enable status code handling.
Enabling the External File Wizards to Retrieve Files Using FTP or HTTP	Enable the external file wizards to retrieve files using FTP or HTTP.

All of these topics are covered in the remaining sections of this chapter.

Connecting to SAS Servers

Overview of SAS Server Connections

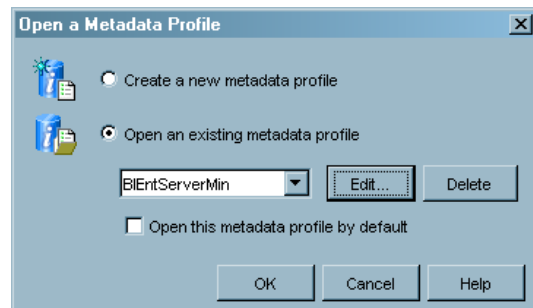
In order to use SAS Data Integration Studio, data integration specialists must be able to connect to your system's SAS Metadata Server and to one or more SAS Workspace Servers. Access to the SAS Metadata Server is necessary because the objects with which users interact directly—libraries, tables, and jobs—are represented by metadata objects. Without access to the SAS Metadata Server, these users cannot do any work. Similarly, users need access to SAS Workspace Servers for many ordinary tasks, such as registering a data source in a metadata repository and running jobs that extract, transform, and load data.

Connecting to a Metadata Server

The easiest way to ensure that data integration specialists can connect to a running SAS Metadata Server is to use SAS Data Integration Studio to open a *metadata profile*, which contains information about the SAS Metadata Server, a metadata repository, and a user. When SAS Data Integration Studio opens this profile, it attempts to connect to the SAS Metadata Server. If it connects successfully, SAS Data Integration Studio will complete its initialization, and the object trees in the interface—such as the Inventory tree—will be populated with selected objects from a metadata repository.

There are two cases that might confront you. If you ran the SAS Configuration Wizard (during installation) on the machine from which you will perform the test, a metadata profile will already exist, and you can use it. Otherwise, you will have to create a metadata profile and then open it.

In either case, you need to go to a machine where SAS Data Integration Studio has been installed and start the application. You do this by selecting **Start ► Programs ► SAS ► SAS Data Integration Studio 9.1**. As the application starts, you will see the Open a Metadata Profile dialog box.



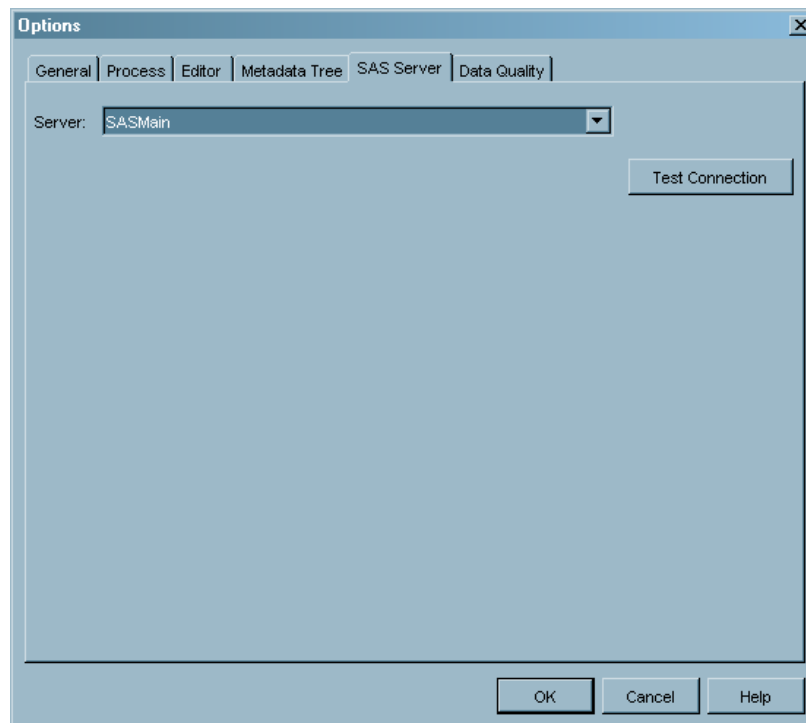
If a metadata profile has been created on this machine, you will see its name listed in the **Open an existing metadata profile** list box. The profile will have the same name as your configuration directory and will have been defined to connect to the SAS Metadata Server using the SAS Administrator account. In this case, click **OK** in the dialog box. You will be prompted for a password. Enter the password for the SAS Administrator account, and click **OK**. If SAS Data Integration Studio is able to connect to the SAS Metadata Server, it will read metadata from a repository and display a set of metadata objects in its tree views.

If no metadata profile has been created, you will have to define one and use it to connect to the SAS Metadata Server. In this case, the **Create a new metadata profile** radio button will be selected when the Open a Metadata Profile dialog box appears. Click **OK** to start the Metadata Profile Wizard. Use the wizard and the on-line help, if necessary, to create a profile. (Do not save your password in the profile.) When you have finished defining the profile, SAS Data Integration Studio will automatically try to use the profile to connect to the SAS Metadata Server.

Connecting to a Workspace Server

When you execute a job in SAS Data Integration Studio, the application submits generated SAS code to a SAS Workspace Server, which executes the code. Therefore, it is imperative that an object spawner be up and running and that SAS Data Integration Studio be able to use the spawner to start a SAS Workspace Server. To test a connection to a SAS Workspace Server, follow these steps:

- 1 Select **Tools ► Options**. An Options dialog box appears.
- 2 In the Options dialog box, select the **SAS Server** tab. The Options dialog box should now look like this:



- 3 Select from the list the name of the SAS Application Server to which the SAS Workspace Server belongs.

- 4 Click **Test Connection**. You might be prompted for a user name and password. If you are, enter credentials that will allow you to be authenticated on the host where the SAS Workspace Server is running. If all goes well, you will see an Information dialog box that says, “Connection to the server was successful.”

Connecting to Data Servers

After you have established that your users can connect to the SAS Metadata Server and the system’s SAS Workspace Servers (as described in “Connecting to SAS Servers” on page 8), it is a good idea to make sure that users can get to the data sources that will provide the input to SAS Data Integration Studio jobs. Here are some examples of these sources:

- DB2 tables
- Sybase tables
- Teradata tables
- ODBC data sources
- Oracle tables
- SAS data sets
- SAS Scalable Performance Data Engine tables

The general procedure for performing this test is to follow these steps:

- Register your data sources in your foundation metadata repository as explained in Connecting to Common Data Sources.
- Use SAS Data Integration Studio’s View Data feature to make sure that you can read data from your different data sources.

After you have registered your SAS tables and DBMS tables in the metadata, these tables will appear in the Tables folder of SAS Data Integration Studio’s Inventory tree.

To determine whether your data integration developers will be able to read data from a particular data server, perform these steps:

- 1 In the Inventory tree, select a table that is managed by that server.
- 2 Select **View ► View Data**.

A View Data window appears and shows you the data in the table.

#	CHECKING_ID	CHECKING_PRIMARY_CLIENT_ID	CHECKING_CLIENT_TYPE_CD	CHECKING_BRANCH_SORT_CD
1	CHK-197	CHK0002314814	OW	RVAFNY
2	CHK-198	CHK0001712332	NS	YMFOSY
3	CHK-199	CHK0002069310	IR	NMPVJJ
4	CHK-200	CHK0002369415	RT	DGPSIL
5	CHK-190	CHK0001886954	GD	YVLBXS
6	CHK-191	CHK0002651003	FI	SFTVHC
7	CHK-192	CHK0002329076	OW	XNRYLJ
8	CHK-193	CHK000232444	AU	MUHHKD
9	CHK-194	CHK0000922871	VN	LTMUCJ
10	CHK-195	CHK0000589705	SW	BOYCTN
11	CHK-196	CHK0002067258	YC	OFMHGW
12	CHK-183	CHK0001191514	LD	MSQSLJ
13	CHK-184	CHK0002477309	HT	MJGHH
14	CHK-185	CHK0002032830	NC	DFGBVJ
15	CHK-186	CHK0000200032	PX	TNCGTI
16	CHK-187	CHK0000408579	LU	ERTJVH
17	CHK-188	CHK0002481443	GZ	YZDXGP
18	CHK-189	CHK0001516936	IC	QINECD
19	CHK-177	CHK0001589248	YD	EMHKG
20	CHK-178	CHK0001820634	VWZ	RGTRFG

Note: You can quickly determine the type of a table in the Tables folder by bringing up the table's Properties dialog box and selecting the **Physical Storage** tab. The **DBMS** list box will contain a value such as SAS, Oracle, or Sybase. △

Setting Up Change Management

Overview of Setting Up Change Management

SAS Data Integration Studio contains a change management feature that enables data integration developers to check metadata objects out of a foundation repository into a work repository, called a project repository. There, a developer can modify the checked-out objects (and create new objects). Meanwhile, the corresponding objects in the foundation repository are locked. When the developer finishes his or her work and checks in any changes, these locks are released. The best practice in this area is for each data integration specialist to have his or her own project repository. This arrangement prevents developers from making changes to the same metadata objects at the same time.

The following list *summarizes* the administrative tasks you must perform to set up change management in the case where there is one project repository per developer. (The sections that follow the list provide details where necessary about how to perform these tasks.)

- 1 Set up an operating system user account for each data integration specialist. On Windows systems, you must give the user—or a group to which the user belongs—the user right “Log on as a batch job.” You should already have performed this task.
- 2 Use the User Manager plug-in to SAS Management Console to create a metadata object that represents the user in the foundation repository. You should already have performed this task as well.
- 3 Create a new directory in the file system that will hold the contents of the new project repository. Make sure that only the user who invokes the SAS Metadata Server has write access to this directory. For details on how to perform this step, see “Setting Up Change Management” on page 11.
- 4 Use SAS Management Console to create the new project repository..
- 5 In SAS Management Console, set the permissions on the foundation and project repositories so that the data integration specialist cannot write data directly to the foundation repository, but can write to his or her project repository.
- 6 Have the data integration specialist create a metadata profile that he or she can use to connect to the SAS Metadata Server and to specify which project repository he or she will be working in.
- 7 Have the data integration specialist use the metadata profile to make sure that change management is working.

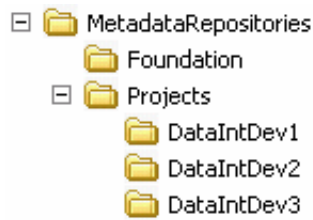
As previously mentioned, you should already have performed steps 1 and 2. If you have not performed those steps, perform steps 1 and 2 now. The other steps are explained in the sections that follow.

Creating a Repository Directory

On the machine where you installed your SAS Metadata Server, go to the directory `SAS-config-dir\Lev1\SASMain\MetadataServer\MetadataRepositories`. There, you

will see a directory for your foundation repository. Add a new directory called **Projects**. Then change directories to the **Projects** directory, and create a directory named for the data integration specialist. This directory will contain the user's project repository. In addition, set the ownership and permissions on this new directory so that only the user who starts the SAS Metadata Server has read and write access to the directory. On UNIX systems, the owner should be **sas**, and the permissions should be set to **700**. On Windows systems where the SAS servers are running as services, **SYSTEM** should have full control of the directory. (On Windows systems where the servers are started using scripts, the user who runs the script **startMetadataServer.bat** should have full control of the directory.)

When you have created a project repository directory for each data integration developer, your directory structure should look something like this:



Creating a Project Repository

Before you begin this step, make sure that you have created a metadata object for the user and the directory that will hold the contents of the new repository. The wizard that you use to create the repository prompts you for information about these items.

Perform these steps to create the repository in SAS Management Console:

- 1 In the left pane, expand the **Metadata Manager** portion of the tree, and select the **Active Server**. Then, select **Actions ► Add Repository**. A wizard that guides you through the process of creating a metadata repository starts.
- 2 In the Select Repository Type window, select the **Project** radio button. Then click **Next**.
- 3 In the General Information window, enter a name for the new repository in the **Name** text box. Entering a description of the repository in the **Description** text box is optional. Click **Next**.
- 4 In the Definition of Data Source window, you are prompted for three pieces of information: an Engine, a Path, and Options.
 - a Engine—Generally, you should accept the default value, **Base**. This setting is appropriate whenever you are storing your metadata in SAS data sets (the default). If you are using a third party DBMS for repository storage, select **DB2** or **Oracle**, as appropriate.
 - b Path—Enter a full path to the directory that will hold the contents of the repository, or use the available **Browse** button to specify this directory.
 - c Options—Do not enter any options.

Click **Next**.

- 5 In the Define Repository Dependencies window, specify that your project repository will depend on the foundation repository. This means that the user will check metadata objects out of the foundation repository into the user's project directory to work on the metadata objects. The user will then check these objects (and any new objects) back into the foundation repository.

To specify this relationship between repositories, select **Foundation** from the list of repositories on the left, and then click the right arrow. The foundation repository icon will move to the **Repository will depend on** list on the right. Then click **Next**.

- 6 In the Choose Repository Owner window, select the user for whom you are creating the project repository, and then click **Next**.
- 7 Finally, in the Current Settings window, review the data you have entered; then click **Finish**.

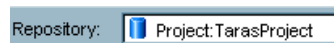
In SAS Management Console, you will see an icon that represents the new repository displayed in the Metadata Manager section of the tree in the left pane.

Setting Metadata Permissions for Your User

After you have created a project repository for a user, you need to use SAS Management Console to set (in metadata) the permissions for that user to access the foundation repository and the permissions for that user to access his or her project repository. The user needs the ReadMetadata and CheckInMetadata permissions for the foundation repository, and ReadMetadata and WriteMetadata permissions for the project repository. The following steps explain how to set up permissions for the owner of the project repository to access that repository. (You set the user's permissions to access the foundation repository in a similar way.)

Note: When you perform this step, you must be logged on to SAS Management Console as an unrestricted user. For example, you could be logged on as the SAS Administrator (sasadm). To make another user an unrestricted user, you must add that user's ID to the file **adminUsers.txt** and prepend an asterisk to that ID. △

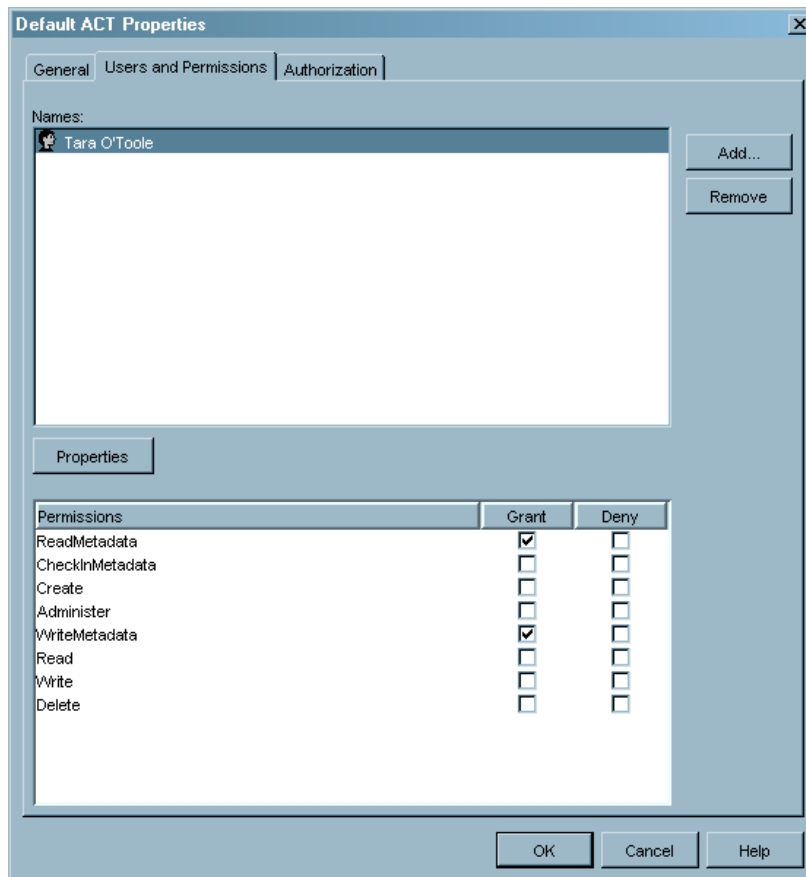
- 1 Select the user's project repository from the **Repository** list box, as shown in the following display.



(This list box is located just below the button bar near the top of the window.)

- 2 In the left pane, expand the **Authorization Manager** section of the tree and, within that section, the **Access Control Templates** section. You will see an icon named Default ACT (Access Control Template). Select this icon and select **File ► Properties**. The Default ACT Properties dialog box appears.
- 3 In this dialog box, click the **Users and Permissions** tab. In the top half of the dialog box, you see a list of users and groups that have permissions defined for this repository. In the lower half of the dialog box, you see the permissions for the currently selected user or group.
- 4 On the **User and Permissions** tab, remove all of the existing user and group names. Then add the name of the owner of the project repository.
- 5 Set the user's permissions as follows:
 - Grant ReadMetadata
 - Grant WriteMetadata.

As this point, your Default ACT Properties dialog box should look something like this:



Note: When you are setting the user's permissions to access the foundation repository, just add the new user and set his or her permissions. Do not remove the existing users and groups. △

- 6 Click **OK**. Now only Tara will be able to work in her project directory.

Creating a Metadata Profile

A metadata profile enables a SAS Data Integration Studio user to connect to a SAS Metadata Server and to specify a default metadata repository (typically the user's project repository). You could set up metadata repositories for all of your data integration specialists, but we recommend that you give users the information necessary to create a profile and let the users create the profiles.

You will need to give each user the following information:

- the full name of the machine on which the SAS Metadata Server is running—for example, server1.na.sas.com
- the port on which the SAS Metadata Server is listening (8561 by default)
- the name of the user's project repository

If you also want to provide instructions about how to create a metadata profile, here they are:

- 1 Start SAS Data Integration Studio. The Open a Metadata Profile dialog box appears.
- 2 In the Open a Metadata Profile dialog box, select the **Create a new metadata profile** radio button; then click **OK**. The Metadata Profile Wizard starts.

- 3 In the Metadata Profile Wizard window, click **Next**. The only purpose of this window is to explain what the wizard does.
- 4 In the Metadata Profile Wizard window, enter a name for your metadata profile in the **Name** text box. You also have the option of selecting the **Open this metadata profile by default** check box. If you will always be working in the same project repository, you should select the check box so that you will not be prompted to select a metadata profile each time that you start SAS Data Integration Studio. If you are working on multiple projects, do not select the check box, so that you can select the profile that you need each time you start the application. After you have supplied this data, click **Next**.
- 5 In the Connection Information window, fill in the following text boxes:
 - Machine**—Enter the full name of the machine on which the SAS Metadata Server is running. (This information is supplied by the administrator.)
 - Port**—Enter the number of the port on which the SAS Metadata Server is listening. (This information is supplied by the administrator.)
 - Username**—Enter your user name. On Windows systems, this name should be of the form *domain\user-name* or *hostname\user-name*.
 - Password**—Enter your password.

We recommend that you do not select the **Save username and password in this profile** check box. If you do, any user can connect to the SAS Metadata Server by simply starting SAS Data Integration Studio on your workstation.

Click **Next**.
- 6 In the Repository Selection window, select your project repository, and click **Next**. (Your administrator will give you the name of this repository.) If no project repository is displayed, see below.
- 7 In the Finish window, click **Finish**.

Using the Metadata Profile

When the user finishes running the Metadata Profile wizard, SAS Data Integration Studio will automatically connect to the SAS Metadata Server and read the appropriate metadata objects. The user will know that he or she is set up correctly for change management if, when the SAS Data Integration Studio interface appears, there is a **Project** tab at the bottom of the tree pane. If the user clicks this tab, he or she will see an icon representing his or her project repository.

Note: If this **Project** tab does not appear, see “Setting Up Change Management” on page 11. △

Responding When No Project Repositories Are Displayed

The most likely reason that a **Project** tab or project repository is not displayed is that the login (user name and password) that is specified in the user’s metadata identity is different from the login that is used to connect to the SAS Metadata Server. You might need to ask your administrator what the correct login is. Other troubleshooting steps you can take include the following:

- Verify that the instructions in “Setting Up Change Management” on page 11 were followed.
- On a Windows platform, verify that the Windows network domain is specified for the login that is part of the user’s metadata identity and for the login that is used to connect to the SAS Metadata Server.

- Verify that operating system security enables the user to read the directory where the Project repository is located.

Note: For information about how to work in a change-managed environment, see the *SAS Data Integration Studio: User's Guide*. Δ

Setting the Default umask on UNIX Systems

When SAS Data Integration Studio writes a target data set, that data set is owned by the user who ran the job that created the data set, and the permissions on the data set file are set like this:

```
rwxr-xr-x
```

These permissions are determined by the SAS Workspace Server's default umask, 022.

This setup can lead to problems if you want multiple data integration specialists to be able to update the same data set. If Developer 1 runs a job that creates a target data set and then goes on vacation, Developer 2 will not be able to run the same job. The job will fail with an error indicating that Developer 2 does not have sufficient authorization to update the target file because the developer does not have write access to the file. To circumvent this problem, you can put your data integration specialists in a group and have the SAS Workspace Server use a umask that sets file permissions so that any member of the group can read and write a file that is created by a member of the group. A umask setting of 002 will enable your developers to share files.

To change the SAS Workspace Server's default umask, perform the following steps:

- 1 Create an operating system group, and make this group the primary group for your data integration specialists.
- 2 Edit the file *SAS-config-dir/Lev1/SASMain/sas.sh* so that the script performs the following actions:
 - a Obtains the user's primary group number.
 - b If the user's primary group is the data-integration-specialist group, sets the umask to 002.

The code below shows the lines that you might add to the *sas.sh* file on an AIX system:

```
CMD=/usr/bin/id
CURR_GID=eval $CMD -g
GID=201
if [ $CURR_GID -eq $GID ]; then umask 002 fi
```

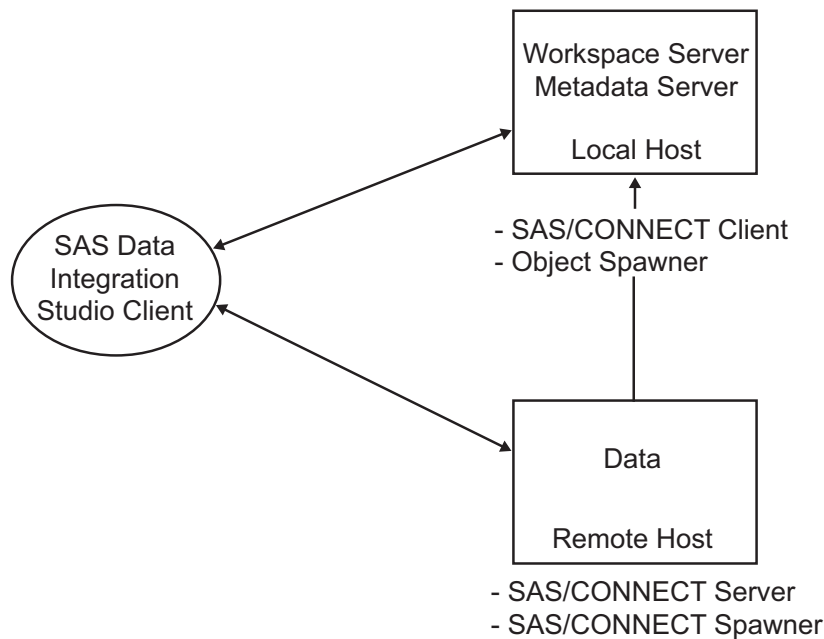
Setting Up Multi-Tier Environments

Accessing Data Remotely

Overview of Remote Data Access

The procedures in this section explain how to enable your SAS Data Integration Studio users to access data on a host that is remote from the SAS Workspace Server by installing SAS/CONNECT software on the SAS Workspace Server host and configuring a SAS/CONNECT Server on the remote-data host.

In order to access data that is remote from your SAS Workspace Server, you must run SAS/CONNECT software on your SAS Workspace Server machine (the *local host*) and also on the machine where your data resides (the *remote host*). To understand this setup, consider the following figure.



In this network, we have three machines:

- the machine running the SAS Workspace Server and the SAS Metadata Server (the local host)
- a separate machine with data (the remote host)
- the SAS Data Integration Studio client machine

Setting Up a Local Host

These procedures assume that SAS software has been installed on this machine. Furthermore, the local host machine needs to be set up as follows:

- The local host runs the SAS Workspace Server.
- The SAS Application Server of which the SAS Workspace Server is a component needs to be set as the default SAS Application Server for SAS Data Integration Studio. This is done from the SAS Data Integration Studio client machine. See “Setting Up Multi-Tier Environments” on page 16.
- SAS/CONNECT software needs to be installed on the local host machine. This should have been done at installation time.

Setting Up a Remote Host

Base SAS and SAS/CONNECT software must be installed on the remote host machine, and a SAS/CONNECT Server must be configured. If you have not already installed SAS software, follow the instructions for a Software Index installation in the *SAS Intelligence Platform: Installation Guide*. After installation, run the Configuration Wizard to configure the SAS/CONNECT software. This will create various scripts, including the `InstallConnectServer.bat` script. You use that script to start the SAS/

CONNECT spawner as a service. Running as a service, the spawner is able to start server processes as needed. You can find **InstallConnectServer.bat** in the *path-to-config-dir\Lev1\SASMain\ConnectServer* directory. In a Windows environment, the **InstallConnectServer.bat** script creates a service. A different script named **StartConnectServer.bat** starts the server.

You can make other configuration changes to the remote host from any machine running SAS Management Console.

In the next two sections, you will define the SAS/CONNECT Server and your SAS library in the metadata managed by the SAS Metadata Server.

Setting Up a Remote SAS/CONNECT Server

Use SAS Management Console to define a SAS Application Server that contains a SAS/CONNECT component that will run on the remote host machine. This server is a second SAS Application Server, in addition to the SAS Application Server running on the local machine (SASMain). This new server is necessary because SASMain does not have access to the remote library; the new SAS Application Server does. Furthermore, by detecting the existence of this new server, SAS Data Integration Studio knows to generate the code necessary to download the data from the remote server. Follow these steps to set up the server:

- 1 Start SAS Management Console.
- 2 Right-click **Server Manager** and select **New Server**. The New Server Wizard appears.
- 3 Select **SAS Application Server**. Click **Next**.
- 4 Enter a name for the new server. Click **Next**.
- 5 Accept the default values for the Server Properties. Click **Next**.
- 6 Specify SAS/CONNECT Server as the type of server to add to the SAS Application Server.
- 7 Select the Basic configuration setting method, and enter the following values:
 - Authentication Domain. When your system was originally set up, a single authentication domain (DefaultAuth) was created. The SAS Workspace Server was placed in this domain, and each SAS Data Integration Studio user (in the metadata) was given permission to start the SAS Workspace Server. In your new multi-tier setup, you can continue to use DefaultAuth if the user name and password that are currently in the metadata are also valid on the host where the SAS/CONNECT Server is running. If those credentials aren't valid on that host, you must create a new authentication domain. You must place the SAS/CONNECT Server in that domain, and create a second login (in the metadata) for each SAS Data Integration Studio user (see "Setting Up Multi-Tier Environments" on page 16).
 - Host Name: Enter the name of the remote host machine.
 - Port Number: Enter **7551**.
- 8 Click **Finish**.

Defining a SAS Library

From SAS Management Console, define a SAS library on the remote host machine. These instructions are appropriate for SAS data sets:

- 1 Expand **Data Library Manager**.
- 2 Right-click **SAS Libraries** and select **New Library**.
- 3 Select **SAS Base Engine Library** and click **Next**.

- 4 Enter a name for the new library and click **Next**.
- 5 Enter the library properties:
 - Libref**— Enter an appropriate libref name.
 - Engine**— Select **Base**.
 - Path specification**— Select a path that you have created. Click **Next**.
- 6 Select your SAS Application Server from the list. The library is assigned to the server that you select.
- 7 Click **Finish**.

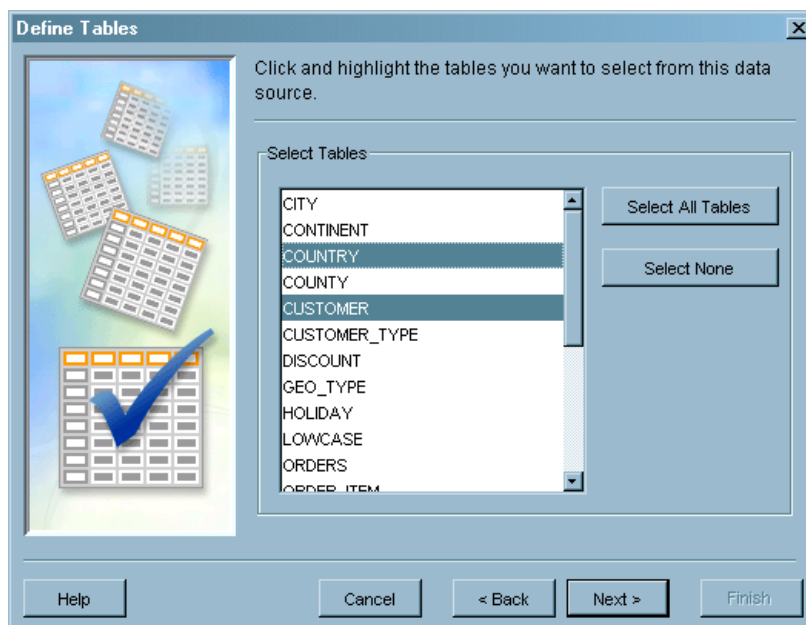
Setting Up a SAS Data Integration Studio Client Machine

From the SAS Data Integration Studio client machine, you need to (1) define a default SAS Application Server, (2) define tables in the metadata for the remote host, and (3) test your setup. In addition, if you have created a new authentication domain for your remote server, you need to create new logins for your SAS Data Integration Studio users. These logins must contain a user ID and password that are valid on the SAS/CONNECT host. See “Setting Up Change Management” on page 11.

Defining Tables in the Metadata for the Remote Host

Follow these steps to define tables in the metadata for the remote host:

- 1 Open SAS Data Integration Studio. If you already have a metadata profile, select it. Otherwise, create a new metadata profile as described at “Setting Up Change Management” on page 11.
- 2 Click **Source Designer**.
- 3 When prompted for the source type that you want to import metadata about, select **Sources** \blacktriangleright **SAS**. Click **Next**.
- 4 On the Select a SAS Library screen, select the library you created earlier. Click **Next**.
- 5 Select one or more tables from the library to register, as in the following screen:

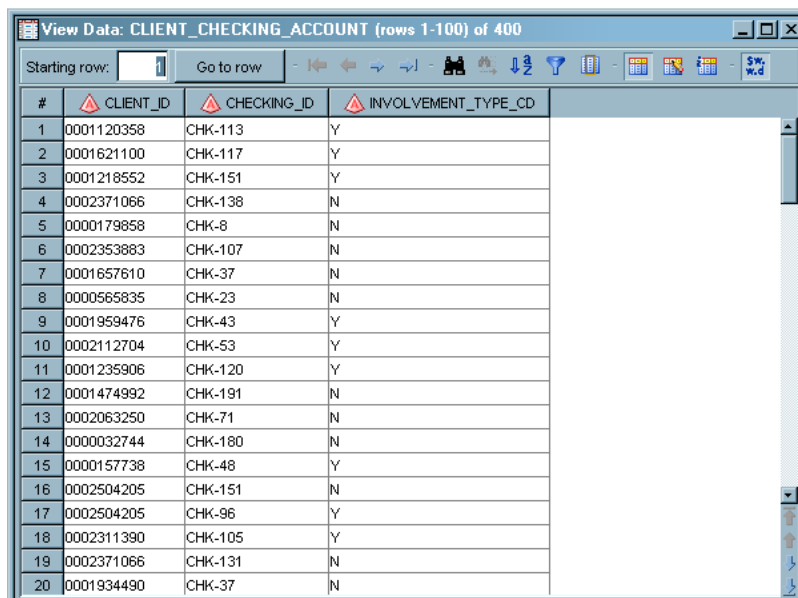


- 6 Select your project repository, click **Next**.

- 7 The Select Group window gives you the opportunity to place the tables. Skip this option and click **Next**.
- 8 Click **Finish** in the Wizard Finish window.

Testing Your Setup Using View Data

Select **Repositories** \blacktriangleright **Foundation** \blacktriangleright **Libraries**. Find and expand your remote library. Right-click one of the tables, and select **view Data**. The table contents appear as shown below:



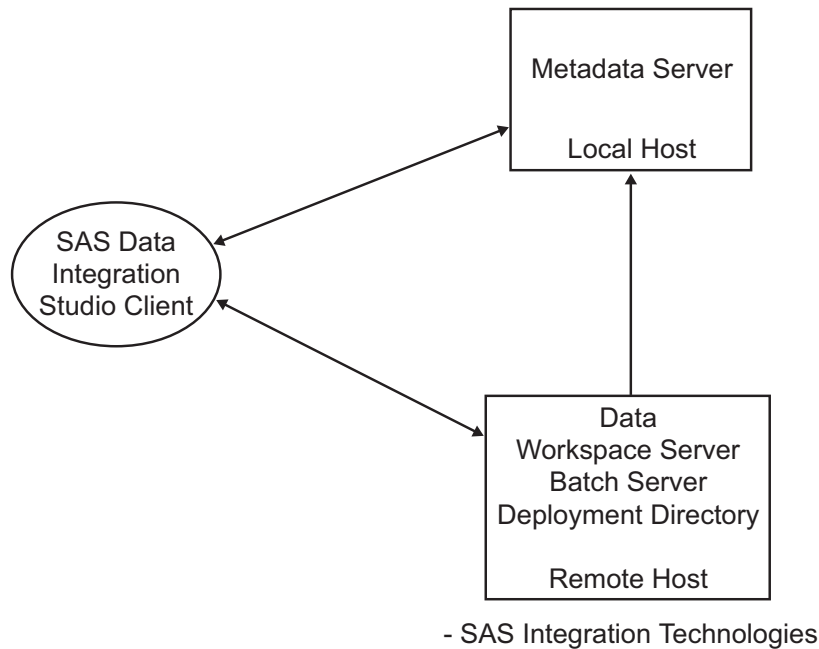
#	CLIENT_ID	CHECKING_ID	INVOLVEMENT_TYPE_CD
1	0001120358	CHK-113	Y
2	0001621100	CHK-117	Y
3	0001218552	CHK-151	Y
4	0002371066	CHK-138	N
5	0000179858	CHK-8	N
6	0002353883	CHK-107	N
7	0001657610	CHK-37	N
8	0000565835	CHK-23	N
9	0001959476	CHK-43	Y
10	0002112704	CHK-53	Y
11	0001235906	CHK-120	Y
12	0001474992	CHK-191	N
13	0002063250	CHK-71	N
14	0000032744	CHK-180	N
15	0000157738	CHK-48	Y
16	0002504205	CHK-151	N
17	0002504205	CHK-96	Y
18	0002311390	CHK-105	Y
19	0002371066	CHK-131	N
20	0001934490	CHK-37	N

This test verifies that you can access your remote library.

Processing Jobs Remotely

Overview of Processing Jobs Remotely

At times you might want to process one or more SAS Data Integration Studio jobs on a large amount of data on a remote machine and then save your results to that remote machine. In such an instance, it might be more efficient to move the job itself to the remote machine. A multi-tiered configuration can make that possible.



To move a job to a remote machine, you need to have a SAS Workspace Server on the remote host. The following software must be installed on the remote host machine:

- Base SAS software
- SAS Integration Technologies software

If you have not already installed this software, follow the instructions for a Software Index installation in the *SAS Intelligence Platform: Installation Guide*. After installation, run the Configuration Wizard to configure the following:

- SAS Object Spawner
- SAS Workspace Server
- SAS Data Step Batch Server

The Configuration Wizard will create a file named `instructions.html` that will give you specific manual instructions to follow in order to complete the process. The steps in that file will be tailored to your own setup and will be based upon your answers to prompts during installation. Below is a generic summary of those steps. Note that in a few cases involving the name of the authentication domain, you might need to enter a different value from that prescribed in `instructions.html`.

Starting the SAS Management Console

SAS Management Console is used to manage your SAS environment and will be used to complete many of the following steps.

The SAS Configuration Wizard will create a new metadata profile and start SAS Management Console. If SAS Management Console does not start automatically, you can start it manually.

Defining Your Application Server and Workspace Server

Using SAS Management Console, you need to define your servers in the metadata. To do this, you use the Server Manager plug-in.

- 1 Right-click **Server Manager** and select **New Server**.
- 2 Under **SAS Servers**, select **SAS Application Server**. Click **Next**.

- 3 Enter a name for the server. (Description is optional.) Click **Next**.
- 4 Accept the default values for software versions. Click **Next**.
- 5 Select **Workspace Server**.
- 6 Select **Basic Configuration**, and enter the following values:
 - Authentication Domain: In some cases this will be your default authentication domain **DefaultAuth**. However, in certain circumstances (described in “Setting Up Multi-Tier Environments” on page 16), you might need to enter a different authentication domain.
 - Host Name: Enter the name of your remote host.
 - Port Number: Enter **8591**.
- 7 Review settings and select **Finish**.

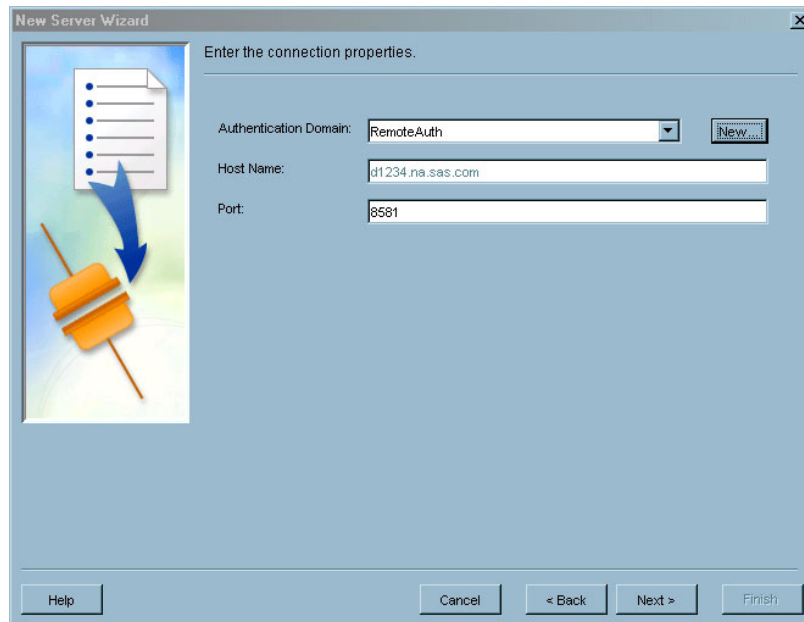
Editing the SAS Command for the Workspace Server

- 1 Select + to expand the **Server Manager** node. Fully expand all three levels of the SAS Application Server you created previously.
- 2 Highlight the **Workspace Server** (lowest) level. Using the right mouse button, select **Properties**.
- 3 On the **Options** tab, replace the existing text in the **Command** field with the command provided in **instructions.html**.
- 4 Click **OK** to save changes.

Defining Your SAS Object Spawner

The SAS Object Spawner is used to start SAS Workspace Server processes.

- 1 Right-click **Server Manager** and select **New Server**.
- 2 Under **SAS Servers - Spawners**, select **Object Spawner**. Click **Next**.
- 3 Enter a name for the spawner. (Description is optional.) Click **Next**.
- 4 Accept the default values for software versions. For the **Associated Machine**, select the name of the remote host machine.
- 5 Accept the default values for **Operator Login** and **Log file**. Click **Next**.
- 6 In the list of servers, move the name of the remote SAS Workspace Server to the **Selected** list. Click **Next**.
- 7 Select **Operator Connection**.
- 8 Enter the following connection properties:
 - Authentication Domain: In some cases this will be your default authentication domain **DefaultAuth**. However, in certain circumstances (described in “Setting Up Multi-Tier Environments” on page 16), you might need to enter a different authentication domain.
 - Host Name: Enter the name of the remote host.
 - Port Number: Enter **8581**.



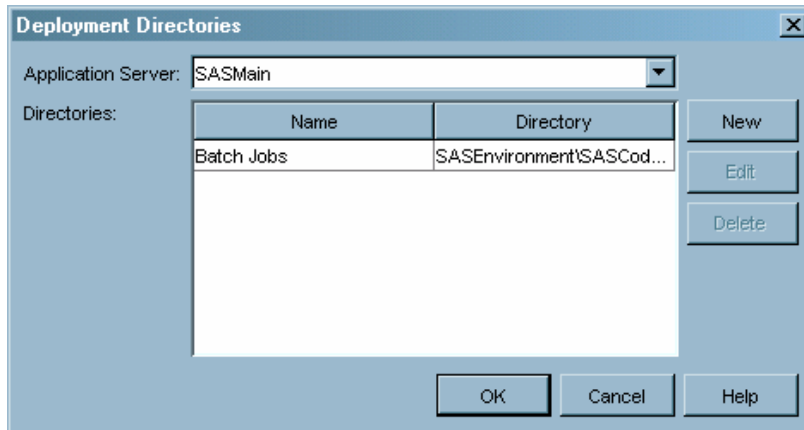
- 9 Review settings and select **Finish**.

Defining Your SAS Data Step Batch Server

- 1 Select the top level of the SAS Application Server you created in the previous procedure.
- 2 Using the right mouse button, select **Add Application Server Component**.
- 3 Select **SAS Data Step Batch Server**. Click **Next**.
- 4 Enter a name for the server. (Description is optional.) Click **Next**.
- 5 Enter the following server properties:
 - Associated Machine: Enter the name of your remote host.
 - Command Line: Enter the command line provided by **instructions.html**.
 - Logs Directory: Enter the directory provided by **instructions.html**.
- 6 Review settings and select **Finish**.

Defining Your Batch Job Deployment Directory

- 1 Highlight **Schedule Manager**. Using the right mouse button, select **Deployment Directories**.
- 2 With the name of the new SAS Application Server selected, select **New**.
- 3 Enter the following directory properties:
 - Name: Enter **Batch Jobs**.
 - Directory: Enter **SASEnvironment\SASCode\Jobs**.



- 4 Select **OK** to create the new directory definition.
- 5 Select **OK** to save your changes.

Verifying Your Object Spawner

To verify the status of the Object Spawner, view the log file, which is located in a path on your system similar to the following:

C:\SAS\EntBIServer\Lev1\SASMainonD9576\ObjectSpawner\logs\objspawn.log.

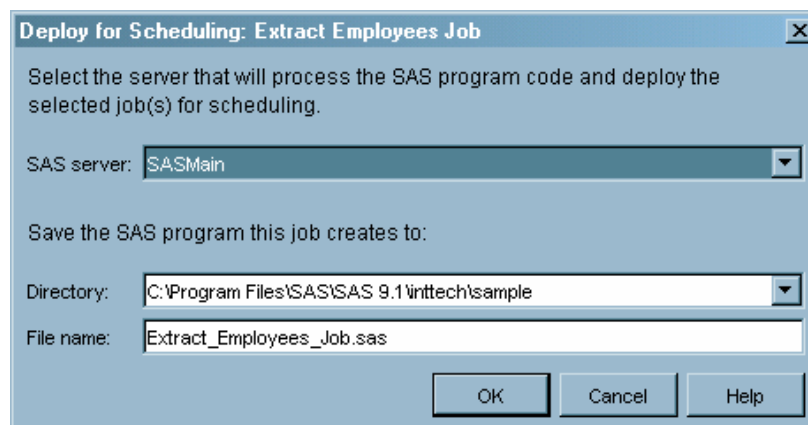
Verifying That the Object Spawner Is Working Properly

- 1 In SAS Management Console, select + to expand the **Server Manager** node. Fully expand all three levels of the SAS Application Server created previously.
- 2 Highlight the **Workspace Server** (lowest) level.
- 3 In the right-hand view, you should see a single connection defined. Highlight that connection name and use the right mouse button to select **Test Connection**.
- 4 Enter a valid user ID and password (for example, *d1234\sasdemo*) and select **OK**. You should see a “Test Connection Successful” message.

Deploying a Job Remotely Using SAS Data Integration Studio

Follow these steps to test the setup you have created:

- 1 Start SAS Data Integration Studio.
- 2 Select **Repositories ► Foundation ► Jobs** in the Inventory tree.
- 3 Right-click the job you that want to deploy and select **Deploy for Scheduling**.



- 4 In the **SAS Server** drop-down list, select the SAS Application Server containing the servers on the remote host.
- 5 Click **OK** to deploy the job.

The deployed code will be automatically copied to the remote host and will be ready for scheduling or running at any time. You must have a Platform Computing Scheduling Server installed before you can schedule a job.

Using Custom Tree Folders for Security

Overview of Custom Tree Folders

When your data integration developers are working in SAS Data Integration Studio, by default the left pane in the main application window displays an Inventory tree. This tree contains a set of folders, each of which corresponds to a type of metadata object that the data integration developer will use routinely in his or her work. The Inventory folder contains the following folders:

- Cubes
- Documents
- External Tables
- Jobs
- Libraries
- Notes
- OLAP Schema
- Tables

By opening a folder, the data integration developer can see the objects of a particular type that are stored in the metadata repository that is specified in his or her metadata profile.

The developer also has the option of creating a Custom tree, which contains user-defined folders. This option has two advantages, one for the data integration developer and one for the administrator.

- The Custom tree enables developers to organize their metadata objects in any manner they see fit. For example, they might want to create a folder that contains all of the metadata objects that are related to a particular set of jobs.
- The Custom tree also displays in the SAS Management Console's **Authorization Manager**. From the **Authorization Manager**, you can set permissions on folders in the Custom tree to specify who can access the objects in the folders and what permissions those users have.

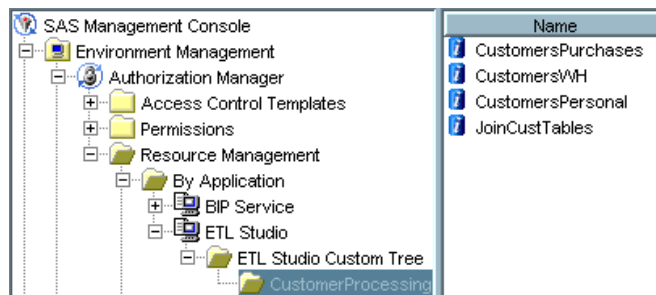
Setting Permissions on Custom Tree Folders

Assume that a SAS Data Integration Studio developer has set up a group (a folder) in the Custom tree to hold the metadata objects with which he or she is working. This section explains how you can use the Authorization Manager in SAS Management Console to set permissions on the group so that only that developer can work with those metadata objects. Perform these steps to set up this type of security:

- 1 In SAS Management Console, expand the following sections of the SAS Management Console tree, in this order:

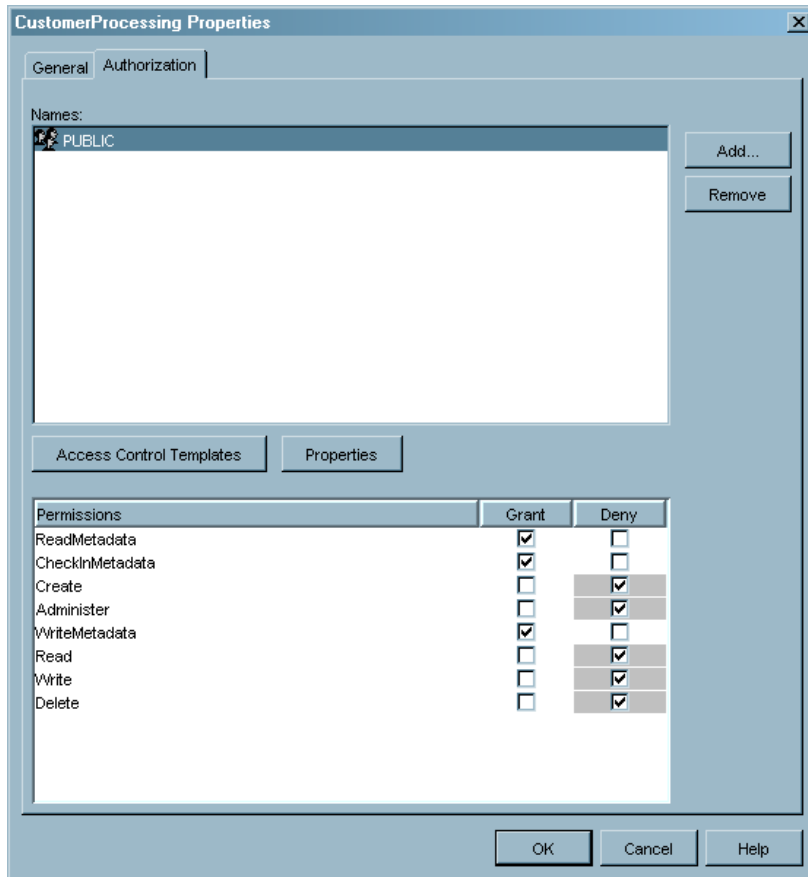
- Authorization Manager**
- Resource Management**
- By Application**
- ETL Studio**
- ETL Studio Custom Tree**

At this point, you will see the top-level groups in the Custom tree, and if you select a group/folder, you will see the objects in that folder in the pane to the right of the tree structure.



In this case, a user has created a group/folder called CustomerProcessing, and this group contains several tables and a job in SAS Data Integration Studio.

- 2 Open the Properties dialog box for the group by right-clicking the group and selecting **Properties** from the pop-up menu.
- 3 In the Properties dialog box, select the **Authorization** tab. The exact settings in this box depend on the Default ACT in use. An example is shown in the following display:



- 4 To make the developer who created the group the only person who is able to access the objects in the group, first deny access to users you do not want to access the objects (such as the members of the group PUBLIC). Then, add your user to the list of names and grant that person appropriate access. If you have set up change management—which is a good idea—you should give the user the permissions ReadMetadata and CheckInMetadata. Otherwise, give the user the permissions ReadMetadata and WriteMetadata.

In a few cases, objects do not inherit permissions from their folder. For example, an external table does not inherit permissions in this way. For these objects, you must set the permissions on the object rather than on the folder. The following table indicates which objects inherit permissions from a group.

Table 3.2 Custom Tree Object Types

Object Type	Description	Inherits from Folder
Cubes	OLAP cubes including dimensions, hierarchies, and levels.	Yes
Documents	Web documents that contain information about another object defined in your metadata repository, such as a table or a job.	Yes
External Tables	Flat files.	No

Object Type	Description	Inherits from Folder
Jobs	Processes that create output. In SAS Data Integration Studio, the job is illustrated by a process flow diagram.	Yes
Libraries	Data libraries for either SAS data sets or DBMS tables.	Yes
Notes	Text objects that contain information about another object that is defined in your metadata repository, such as a table or job.	Yes
OLAP Schemas	OLAP schemas associated with OLAP cubes.	No
Tables	SAS data sets and DBMS tables.	Yes

Note: Be sure to read the next section about the multiple inheritance of access controls. If you do not understand how multiple inheritance works, users might be able to access resources to which you denied them access in the Custom tree. Δ

Administering Multiple Inheritance of Access Controls

As is explained in “Inherited Access Controls” in the “Understanding the Metadata Authorization Layer” section in the *SAS Intelligence Platform: Security Administration Guide*, a single resource can inherit permissions from more than one set of parents. For instance, a SAS data set in a custom tree group inherits permissions not only from its group and the group’s parents, but also from a SAS library and its parents.

In this scenario, you might have specified that User A cannot access the contents of the custom tree group; however, if User A has access to the SAS library that contains the data sets in the group, User A will also be able to access the data sets in the library (unless the permissions on the data sets themselves deny access to this user).

To make sure that this situation does not prevent you from setting up your access controls properly, we recommend the following practice. Create a separate SAS library or database library for each set of tables that should be accessed only by a particular user or group of users. In this way, you can set up the same permissions for a custom tree folder and a corresponding library.

Importing and Exporting Generated Transformations

SAS Data Integration Studio ships with a set of transformations that you can use to perform common transformations such as joining data from a set of input tables in a single target table. Some of the transformations use Java code to transform the data, and other transformations use SAS code. The former class are called Java plug-ins, and the latter are called generated transformations. Data integration specialists can write additional Java plug-ins and generated transformations. However, this section deals only with generated transformation, because it is these transformations that you might need to secure.

SAS Data Integration Studio users can use a Transformation Generator wizard to create generated transformations. They can then export these transformations. In their exported state, the transformations are stored as XML files. Other SAS Data Integration Studio users can import these files, at which point those users can employ the generated transformations in their jobs.

The potential problem with this arrangement is this: suppose that multiple data integration specialists are designing these transformations and exporting them to a central repository. Periodically, all of the users import the transformations from the repository to pick up any new transformations. If one user makes a change to an existing generated transformation and other users import that transformation, any jobs written by the other users that make use of that transformation will be changed without their knowledge.

For this reason, it is important for you to set up a central repository (a directory) to which only you have write access. This way, all data integration users can import generated transformations, but only you can write such transformations to the repository. You can disallow changes to existing transformations or let all users know if such a change takes place.

Importing and Exporting Metadata

Requirements for Supporting Metadata

A common requirement for data integration developers is the ability to do the following:

- design a data warehouse using a data modeling tool such as the AllFusion ERwin Data Modeler
- import metadata that describes the data warehouse into a SAS Intelligence system's metadata repository

You can use either SAS Data Integration Studio or SAS Management Console to import metadata of this type. Both applications enable you to export metadata as well.

Managing the Supported Metadata Formats

The SAS Metadata Server enables you to import metadata from a variety of sources and to export metadata in a variety of formats. The server supports the Object Management Group's Common Warehouse Metamodel/XML Metadata Interchange (CWM/XMI) format, the industry standard for data warehouse metadata integration. In addition, by installing the SAS Metadata Bridges (MITI) software, you can import metadata from market-leading design tool and repository vendors. For information about the MITI software, contact your SAS sales representative.

Importing Metadata

The SAS Data Integration Studio Metadata Import Wizard uses converters installed in the **plug-ins** directory (of SAS Data Integration Studio or SAS Management Console) to import metadata. By default, you have a converter that handles metadata stored in a CWM 1.0/XMI document. If you want to import metadata that is stored in other formats, install the appropriate MITI software described in "Importing and Exporting Metadata" on page 29.

The Metadata Import Wizard enables you to import relational data— that is, data from a SAS library or a DBMS schema. The import process ignores any non-relational data. The following list shows the object types you can import:

- CWMRDB.Schema

- CWMRDB.Table
- CWMRDB.View
- CWMRDB.Column
- CWMRDB.SQLDistinctType
- CWMRDB.PrimaryKey
- CWMRDB.UniqueKey
- CWMRDB.ForeignKey
- CWMRDB.SQLIndex

Do either of the following to start the import wizard:

- In SAS Data Integration Studio, connect to the SAS Metadata Server by using a metadata profile that specifies the repository into which you want to import metadata, and select **Tools ► Metadata Importer**.
- In SAS Management Console, right-click the repository into which you want to import metadata (in the Metadata Manager portion on the tree in the left pane), and select **Import Metadata** from the pop-up menu.

Both procedures start the same wizard.

After you have started the wizard, consult the *SAS Management Console: User's Guide* for information about how to run the wizard. This document contains step-by-step instructions on how to import metadata.

Exporting Metadata

To export data from an Open Metadata Architecture metadata repository to a file, you use the Metadata Export Wizard. By default, the Open Metadata Architecture enables you to export metadata to CWM/XMI files. As with the import feature, you can export metadata in other formats by installing the appropriate MITI software in your SAS Data Integration Studio or SAS Management Console **plug-ins** directory.

There are two restrictions on the export function:

- You can export only relational data, such as data from a SAS library or a DBMS schema. The section “Importing and Exporting Metadata” on page 29 lists the types of objects that you can export.
- If you are exporting metadata from a dependent repository—such as a project directory—metadata is not retrieved from the parent(s) of the repository. For example, only tables that use library definitions in the exported repository are exported. Tables that use library definitions in a parent repository are not exported.

Do either of the following to start the export wizard:

- In SAS Data Integration Studio, connect to the SAS Metadata Server using a metadata profile that specifies the repository from which you want to export metadata, and select **Tools ► Metadata Exporter**.
- In SAS Management Console, right-click the repository from which you want to export metadata (in the Metadata Manager portion of the tree in the left pane), and select **Export Metadata** from the pop-up menu.

Both procedures start the same wizard.

After you have started the wizard, consult the *SAS Management Console: User's Guide* for information about how to run the wizard. This document contains step-by-step instructions on how to export metadata.

Administering Message Queues

What Are Message Queues?

Message queues are collections of data objects that enable asynchronous communication between processes. One application writes a message to a queue. Another application reads messages from the queue to begin the next step in a process. Verification processes guarantee that all messages are transmitted without error.

Queue managers handle message transmission and verification.

About the Third-Party Messaging Software

Fundamental message queue technology is provided in the following third-party software:

- Microsoft MSMQ
- IBM WebSphere MQ

You need to install and configure at least one messaging software package before you register metadata and run SAS messaging jobs. To learn more about the third-party messaging software, see “Application Messaging Overview” in the *SAS Integration Technologies: Developer’s Guide* at http://support.sas.com/rnd/itech/doc9/dev_guide/messageq/index.html.

About Message Queues in the SAS Intelligence Platform

In the SAS Intelligence Platform, you register metadata for queues and queue managers by using SAS Management Console after you install and configure third-party messaging software.

After registering metadata, you use the following transformations to create messaging jobs in SAS Data Integration Studio:

- Microsoft Queue Reader
- Microsoft Queue Writer
- Websphere Queue Reader
- Websphere Queue Writer

SAS Data Integration Studio also enables impact analysis and reverse impact analysis on queues, which you can use to track data flow into and out of queues.

The following data can be written to message queues using the queue writer transformation:

- text of length up to 32,767 bytes
- rows from SAS tables, one row per message
- external files

The queue reader transformation can be configured to read a specified number of rows from the message queue.

Requirements for Message Queues

To implement message queuing, you need to install and configure Microsoft MSMQ or WebSphere MQ. Start the installation process by downloading and installing

third-party software on specified SAS Application Servers. In general you can plan to install a queue manager on each host that runs a SAS Workspace Server that will run message queue jobs. If you choose WebSphere MQ, you can install queue managers on hosts other than SAS Workspace Servers.

Hosts that receive third-party messaging software also need to receive Base SAS and SAS Integration Technologies software, release 9.1.3, Service Pack 4 or later.

SAS Intelligence Platform clients that run SAS Data Integration Studio need to install release 3.4 or later. As part of the installation of SAS Data Integration Studio 3.4, you are required to use SAS Management Console to run **Tools ► Update Metadata for SAS Data Integration Studio in SASMC**. If you are using change management at your site, all metadata needs to be checked in before you can run the metadata update. This generally happens as a normal part of the SAS Data Integration Studio installation process.

Installing and Configuring the Microsoft MSMQ Message Queue Software

Complete the following steps to install the MSMQ software on a SAS Workspace Server host that will run message queue jobs:

- 1 Ensure that the host meets the requirements for message queues, as described in “Requirements for Message Queues” on page 31.
- 2 In Windows, select **Add/Remove Programs**.
- 3 In the shortcut bar of the Add/Remove Programs window, select **Add/Remove Windows Components**, which opens the Windows Components Wizard.
- 4 In the Windows Components Wizard, select the **Message Queueing** check box and click the **Details** button, which displays the Message Queueing dialog box.
- 5 In the Message Queueing dialog box, select all four subcomponent check boxes, and then click **OK** to return to the Windows Components Wizard.
- 6 Click **Next** to install the MSMQ software, and then close the Windows Components Wizard.
- 7 Use the MSMQ software to create the queues and queue managers that are required on this particular host. You need to create at least one queue for each queue manager. The same object names will be used in SAS, so you might want to include the host name as part of the object names.
- 8 Use the Microsoft MSMQ software to start the queue managers.
- 9 Use SAS Management Console to register metadata for queues and queue managers, as described in “Registering Metadata for Queues” on page 33.

Installing and Configuring the WebSphere MQ Message Queue Software

Follow these steps to install and configure the WebSphere MQ software:

- 1 Download the software and documentation. Start with the following Web site: <http://www-306.ibm.com/software/integration/wmq/index.html>. Work with installation representatives from IBM and SAS as needed. At minimum, you need to install the WebSphere MQ Client software on all SAS Workspace Server hosts that will run message queue jobs. Make sure that you meet the requirements for message queueing that are specified in “Requirements for Message Queues” on page 31.

Note that if you install only the WebSphere MQ Client software, then you need to configure the environment variable MQSERVER to communicate with the WebSphere MQ Server software.

- 2 Use the WebSphere MQ software to create queues and queue managers on their respective hosts. For each queue manager you need to define at least one queue. For reference in SAS, you might want to include the machine name in the names of the queue managers.
- 3 Use the WebSphere MQ software to start the queue managers on their respective hosts.
- 4 Use SAS Management Console to register metadata for queues and queue managers, as described in the next section.

Registering Metadata for Queues

Complete the following steps to register metadata for a queue and a queue manager:

- 1 In SAS Management Console, right-click **Server Manager** and select **New Server**, which opens the New Server Wizard.
- 2 Under **Queue Managers**, select either **MSMQ Queue Manager** or **Websphere Queue Manager** and click **Next**.
- 3 Enter a name and description for the new queue manager. The name needs to be exactly the same as the name that is defined in the third-party messaging software. The names are case-sensitive. When you are finished, click **Next**.
- 4 Check the queue type and SAS version number, and click **Next**.
- 5 Specify an authentication domain. Click **New** to define a new authentication domain.
- 6 Specify a full network name for the host.
- 7 Specify a port number.
- 8 Register metadata for at least one queue. One queue is required to be registered for each queue manager. Click **New** to display the New Queue dialog box.
- 9 In the New Queue dialog box, enter a queue name, description, and queue manager name. If you are registering metadata for a queue that already exists in the third-party software, the queue name and the queue manager name must match exactly the names that are defined in the third-party messaging software. The names are case-sensitive. Click **OK** to display the queue in the **Selected items** list. Click **Next** to apply your input.
- 10 Review your entries and click **Finish**.

Be sure to register metadata for all of the queues and queue managers that will be used in SAS jobs.

To begin running messaging jobs, make sure that the queue managers are running on their respective hosts.

Managing Queues and Queue Managers

To modify the metadata definitions of queues and queue managers, use the New Server Wizard in SAS Management Console to delete the existing definitions and create new definitions.

Before you delete metadata, you might want to run impact analysis and reverse impact analysis in SAS Data Integration Studio to examine the related jobs. To run impact analysis, right-click the queue in the **Inventory** tab and select **Impact**

Analysis or Reverse Impact Analysis. The resulting analytical report enables you to display the properties of tables, jobs, and transformations.

To delete metadata for queues and queue managers, right-click the object in the Server Manager of SAS Management Console and select **Delete**. Use the third-party software to delete the physical queue objects.

Providing Support for Web Services

Web Services

Beginning in version 3.4, SAS Data Integration Studio supports the creation of stored processes that can be invoked through a Web service. For complete information about the procedure for creating these special stored processes, see the SAS Data Integration Studio help. The interfaces to these stored procedures are published in WSDL (Web Services Description Language), and remote clients both discover the available services and execute services using SOAP requests.

To provide Web services to clients, in addition to having a SAS Metadata Server and a SAS Stored Process Server, you are required to install and configure the SAS BI Web Services for Java or the SAS BI Web Services for .NET. These products handle service requests from remote clients. If you do not have one of the two products installed, you must install one of them, as explained in the next section.

SAS BI Web Services

SAS BI Web Services for Java Versus SAS BI Web Services for .NET

You can use either SAS BI Web Services for .NET or SAS BI Web Services for Java to handle requests for Web services. SAS BI Web Services for .NET is supported on the following platforms: Windows 2000, Windows XP Professional, and Windows 2003. SAS BI Web Services for Java is supported on any operating system that can host one of the SAS Intelligence Platform's supported Java Application Servers. For information on what Java Application Servers are supported, see the Third-Party Software Downloads page at <http://support.sas.com/thirdpartysupport>. Click the Web Application Servers link for your version of the SAS Intelligence platform to see the list of supported servers.

For more information about the differences between the two versions of the SAS BI Web Services, see the section "Deciding Between .NET and Java" in the *SAS Integration Technologies: Developer's Guide*, at http://support.sas.com/rnd/itech/doc9/dev_guide/websrvcs/decide.html.

Install SAS BI Web Services for .NET

To install and configure SAS BI Web Services for .NET, perform the following steps.

- 1 Install and configure the following prerequisite software (in this order):
 - Microsoft Internet Information Services (IIS)
 - Microsoft .NET Framework
- 2 If necessary, add the contents of the CD containing the SAS BI Web Services for .NET to your SAS Software Depot.

Note: You can find detailed instructions for performing this step and the following steps in the *SAS Intelligence Platform: Installation Guide*. △

- 3 If you received a new SAS Installation Data (SID) file when you ordered the SAS BI Web Services for .NET, put that SID file in your project directory.
- 4 Install the BI Web Services for .NET using the SAS Software Navigator's Software Index option. For details about how to perform this type of installation, see "Install Third-Party Software Using Software Index" in the chapter "Installing Third-Party Products" in *SAS Intelligence Platform: Installation Guide*. (You add a SAS product to an existing system in the same way that you install some third-party products.)
- 5 Use the SAS Configuration Wizard to configure the SAS BI Web Services for .NET. For information about using this wizard to configure a single product, see "Configure a Newly Added Product" in the chapter "Software Index Installations" in the *SAS Intelligence Platform: Installation Guide*.

Install SAS BI Web Services for Java

SAS BI Web Services for a Java is a J2EE Web application that runs on a Java Application Server. It also has a dependency on the SAS Web Infrastructure Kit. Therefore, unless you already have one or more of these products, you must install the following items:

- a Java 2 SDK
- a supported servlet container or Java Application Server
- the SAS Web Infrastructure Kit
- the SAS BI Web Services for Java

To install and configure these products, perform the following steps:

- 1 Install the Java 2 SDK.
 - a Go to the Third-Party Software Downloads page (<http://support.sas.com/thirdpartysupport>).
 - b Locate the set of links that is appropriate for your version of the SAS Intelligence Platform, and click the Java Development Kits link. You will see a list of kits organized by operating system.
 - c Search for the row that has your operating system in the Platform column, and click the corresponding Download link. This will take you to a third-party Web site where you can download the Java 2 SDK.
 - d Download the SDK, and run the installation program.
- 2 Install a servlet container or Java Application Server.
 - a If you have not already done so, determine which product you want to use. To see what products are supported, go to the Third-Party Software Downloads page (<http://support.sas.com/thirdpartysupport>), locate the set of links that is appropriate for your version of the SAS Intelligence Platform, and click the Web Application Server link. You will see a table that lists the supported servers.

Note: SAS BI Web Services for Java is not currently supported on WebSphere. △

- b Obtain the server from the vendor. You can download Tomcat from the Apache Web site for free by using the link on the Third-Party Software Downloads site. You can purchase WebLogic from BEA Systems (<http://www.bea.com>).
- c Install the server. Installation instructions for installing WebLogic are available from the Web page mentioned in step a.

- 3 If necessary, add the contents of the CDs containing the SAS Web Infrastructure Kit and SAS BI Web Services for Java to your SAS Software Depot.
Note: You can find detailed instructions for performing this step and the following steps in the *SAS Intelligence Platform: Installation Guide*. \triangle
- 4 If you received a new SAS Installation Data (SID) file when you ordered the products mentioned in the previous step, put that SID file in your project directory.
- 5 Install the SAS Web Infrastructure Kit and the SAS BI Web Services for Java using the SAS Software Navigator's Software Index option. For details about how to perform this type of installation, see "Install Third-Party Software Using Software Index" in the chapter "Installing Third-Party Products" in *SAS Intelligence Platform: Installation Guide*. (You add SAS products to an existing system in the same way that you install some third-party products.)
- 6 Use the SAS Configuration Wizard to configure the SAS Web Infrastructure Kit and SAS BI Web Services for Java. For information about using this wizard to configure these products, see "Configure a Newly Added Product" in the chapter "Software Index Installations" in *SAS Intelligence Platform: Installation Guide*. The SAS Configuration Wizard will guide you through the process of defining the necessary metadata objects and deploying your Web applications to your Java Application Server.

After you have installed and configured the SAS BI Web Services for Java, a README file (`xmla_readme.html`) will be available in the directory `SAS-install-dir\Web\WebServicesforJava\1.0`. Read this document for further information about the product.

Calling Web Services

There are a couple of sources of information about writing clients that can access your Web services. One source is the *SAS Integration Technologies: Developer's Guide*. See the section of the document called "SAS BI Web Services" (http://support.sas.com/rnd/itech/doc9/dev_guide/websrvcs/index.html). If you have installed the SAS BI Web Services for Java, you can also find information in the README file `SAS-install-dir\Web\WebServicesforJava\1.0\xmla_readme.html`.

Enabling the Bulk Loading of Data

Bulk Loading Data into DBMS Tables

When data integration specialists use the SQL Join transformation to write data to tables in a DBMS, they have the option of specifying that they want to use the DBMS's bulk-loading capabilities. When this option is used, entire rows of data are written to the database at once, which can significantly improve performance.

By default, however, your object spawner is configured so that operating system commands cannot be executed from a spawned SAS Workspace Server, which is a requirement for using the bulk-load feature. To reconfigure your object spawner, follow the directions in the next section. Also, note the caution at the end of the procedure.

Enable the Workspace Server to Execute Operating System Commands

To enable a SAS Workspace Server to execute operating system commands, you must add the `-allowxcmd` spawner invocation option to the command that starts the object spawner. Use the following directions, as appropriate for your operating system.

On Windows systems, perform these steps:

- 1 In a Command Prompt window, change directories to `SAS-config-dir\Lev1\SASMain\ObjectSpawner`.
- 2 Stop the object spawner by entering the command `ObjectSpawner stop`.
- 3 If the object spawner is set up to run as a service, remove the service. You can do this by entering the command `ObjectSpawner remove`.
- 4 Edit the script that is used to start the object spawner: `SAS-config-dir\Lev1\SASMain\ObjectSpawner\ObjectSpawner.bat`. Add the spawner invocation option `-allowxcmd` to the appropriate line. If you will run the object spawner as a service (which is recommended), append the option to the line following the label `:install`. Otherwise, append the option to the line following the label `:start2`.
- 5 Save your changes.
- 6 If you want to run the object spawner as a service, install the service. You can do this by entering the command `ObjectSpawner install`.
- 7 Start the object spawner by entering the command `ObjectSpawner start`.

On UNIX systems, perform these steps:

- 1 Change directories to `SAS-config-dir/Lev1/SASMain/ObjectSpawner`.
- 2 Stop the object spawner by entering the command `ObjectSpawner.sh stop`.
- 3 Edit the script `ObjectSpawner.sh` so that the command used to start the object spawner contains the option `-allowxcmd`. Add the option to the line that begins with `$CMD`, just before the first `>` sign.
- 4 Save your changes.
- 5 Restart the object spawner by entering the command `ObjectSpawner.sh start`.

CAUTION:

After you have loaded your data, you should remove the `-allowxcmd` option from your object-spawner script and restart the spawner. There are security risks associated with allowing system commands to be executed from a client application. Clients can use host commands to perform potentially harmful operations, such as file deletion. △

Testing the Platform Computing Scheduling Server

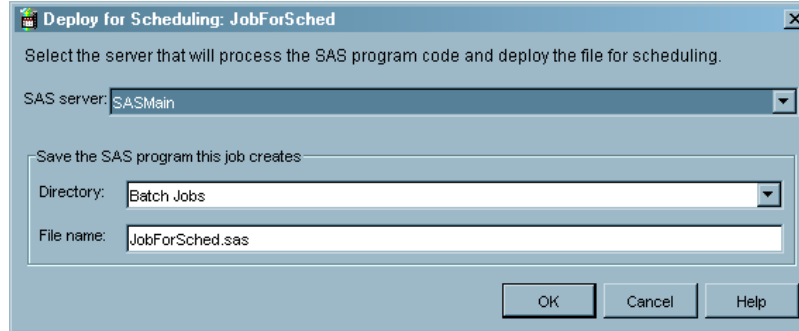
Data integration specialists can create sets of SAS Data Integration Studio jobs, called flows. Each job within a flow can be scheduled to execute as follows:

- at a certain time
- at a time that depends on the state of the file system
- at a time that depends on the status of another job within the flow

The software that enables this scheduling includes the Schedule Manager plug-in to SAS Management Console and a product from Platform Computing: a Platform Computing Scheduling Server. During the initial installation of your system, you will have installed these products and configured the necessary batch and scheduling servers. Now you need to test the scheduling system.

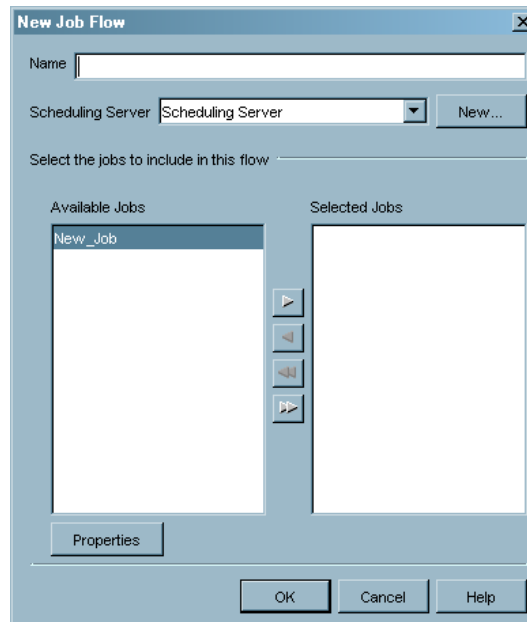
Here is how to quickly make sure that the scheduler is working correctly.

- 1 Create an empty job in SAS Data Integration Studio. You can do this by starting the New Job Wizard, entering a name (such as JobForSched) in the **Name** text box, and clicking **Finish**. A new job appears in the Jobs folder of the Inventory tree.
- 2 Deploy the new job for scheduling by right-clicking the icon for the new job and selecting **Deploy for Scheduling**. The Deploy for Scheduling dialog box appears.



From the **SAS server** list box, select the SAS Application Server that contains your batch server, and from the **Directory** list box, select the name that you specified earlier (**SASenvironment\SASCode\Jobs** by default) for your deployment directory. Use the default value for **File name**. Then click **OK**.

- 3 In SAS Management Console, use these steps to create a job flow (a set of related jobs):
 - a Right-click the Schedule Manager, and select **New Flow** from the pop-up menu. The New Job Flow dialog box appears.



- b Enter a unique name for the job flow in the **Name** text box.
- c From the **Scheduling Server** list box, select the name of the Platform Job Scheduling Server that you created earlier.
- d Move the empty job you just created from the list of available jobs to the list of selected jobs.

- e Click **OK**.

A job flow icon appears beneath the Schedule Manager.

- 4 In SAS Management Console, schedule the new job flow that you want to run:
 - a Right-click the Job Flow icon, and select **Schedule Flow** from the pop-up menu that appears. A Schedule Flow dialog box appears.
 - b In the Schedule Flow dialog box, leave the value of the **Trigger** list box set to **Run Once** and click **OK**.

You should see a message indicating that the flow has been scheduled to run.

- 5 To verify that the job ran successfully, use the Flow Manager application (part of the Platform Computing Scheduling Server).

Setting Up the SAS Data Quality Server Software

Overview of SAS Data Quality Server Software

By installing the SAS Data Quality Server software and configuring a SAS Application Server to read a Quality Knowledge Base, you enable data integration developers to use the data quality transformations **Create Match Code** and **Apply Lookup Standardization**. Those developers can also make use of the data quality functions that are provided in the Expression Builder, which is available in many of the data integration transformations.

This section explains the installed configuration of the SAS Data Quality Server software, and how to perform a simple test to ensure that the system is working. This section also covers several administrative tasks associated with data quality, including the following:

- downloading new locales
- creating new schemes
- setting data quality options in SAS Data Integration Studio

Note: If you are unfamiliar with the subject of data quality and the terminology used in this section, see the *SAS Data Quality Server: Reference*. This document is available in the SAS Help and Documentation and on the SAS OnlineDoc CD-ROM. Δ

Understanding the Data Quality Configuration

If you have installed the SAS Foundation software and the SAS Data Quality Server software and have run the SAS Configuration Wizard, everything should be set up for your data integration developers to use the data quality transformations and functions. This automatic setup is convenient; however, you need to understand how things are set up in case you need to make changes later.

During installation, the SAS Data Quality Server software is installed in *sas-root\dquality*. A typical default path might be **C:\Program Files\SAS\SAS 9.1\dquality**.

Locales and schemes are located in directories subordinate to **dquality**: **sasmisc\content\locale** and **sasmisc\content\scheme**.

A data quality configuration file is used to specify the location of the Quality Knowledge Base. When you start a local interactive SAS session, the data quality configuration is referenced by default at *sas-root\dquality\sasmisc\dqsetup.txt*.

When you use a client application such as SAS Data Integration Studio to start a SAS Workspace server session on a SAS Application Server, the data quality configuration file that is referenced is located by default at *app-server-root\Lev1\SASMain\dqsetup.txt*. A typical default path might be **C:\SAS\EGServers\Lev1\SASMain\dqsetup.txt**, where SASMain is the name of the SAS Application Server.

On SAS Application Servers, the location of the data quality configuration file is specified in the SAS configuration file *app-server-root\Lev1\SASMain\sasv9.cfg*. In the SAS configuration file, the line that specifies the location of the setup file (by default) is as follows:

```
-dqsetuploc "dqsetup.txt"
```

In that same SAS configuration file, the server's default data quality locale is specified as:

```
-dqlocale (ENUSA)
```

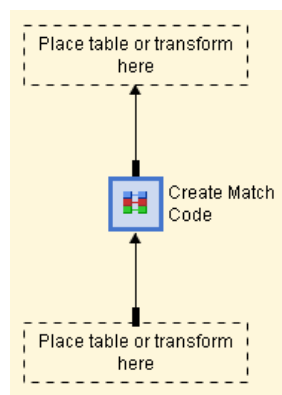
You can change values in the SAS configuration file without restarting the SAS Application Server. Any Data Integration Studio clients will need to be restarted in order to use any new values.

You can use the Properties windows of the data quality transformations to override the server defaults for data quality. You can also use the **Data Quality** tab of the Options dialog box in SAS Data Integration Studio to override server defaults in metadata profiles. To display the Options dialog box, select **Tools ► Options**.

Testing the SAS Data Quality Server Software

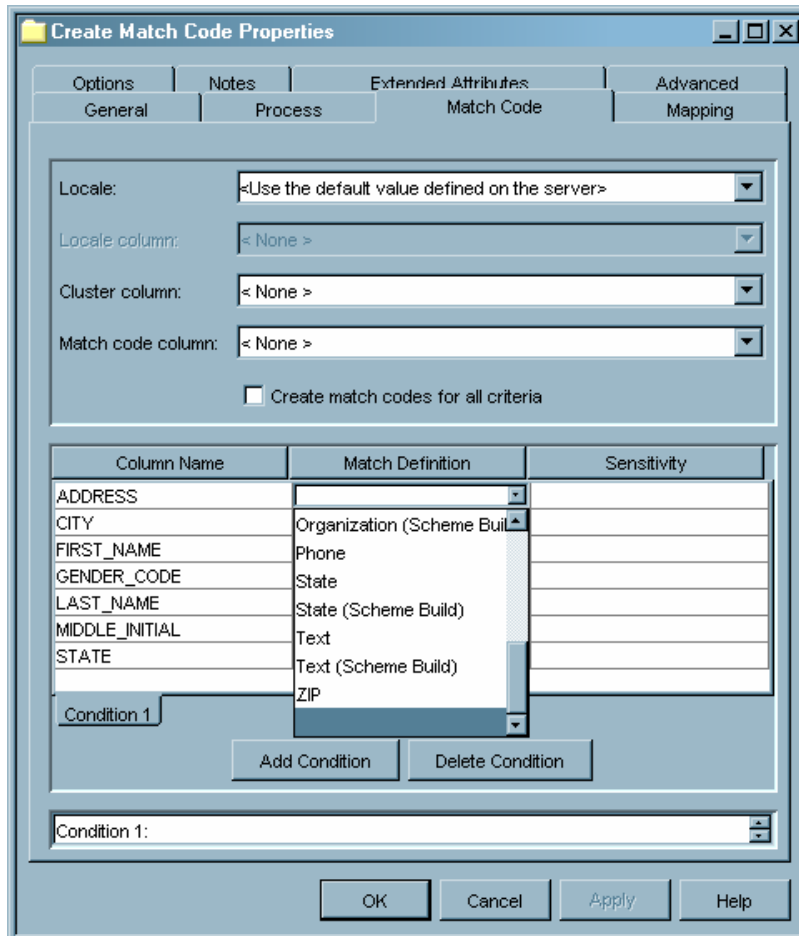
A simple procedure to verify that your SAS Data Quality Server software is working is to create a job in SAS Data Integration Studio that contains a Create Match Code transformation. Follow these steps to create such a job.

- 1 From the SAS Data Integration Studio desktop, select **Tools ► Process Designer** to start the New Job Wizard.
- 2 In the New Job Wizard, enter a name for the job—such as Create Database Match Codes—in the **Name** field. Then, click **Finish**. A new Process Designer window appears on the right side of your workspace.
- 3 From the **Process Library** tree, select and drag the Create Match Code template into the Process Designer. You should see the following template in the designer:



- 4 From the **Inventory** tree, or another tree view, select and drag the metadata object for any table to the source drop zone.

- 5 From the **Inventory** tree, or another tree view, select and drag the metadata object for any table to the target drop zone. Both a Loader and the target table will be added to the graphical representation of the job.
- 6 Double-click the **Create Match Code** transformation to display the **Properties** menu.
- 7 As the Properties window opens, a dialog box indicates that match definitions are being loaded from the Quality Knowledge Base. This indicates that the SAS Data Quality Server software has been properly installed and configured.
- 8 To see a list of the match definitions that were loaded, specify a source table for the job and display the **Match Code** tab of the Properties window. Double-click the **Match Definitions** column to show the list of match definitions, as shown in the following display:



Downloading Locales

When initially installed, the Quality Knowledge Base contains a single locale (English/USA). You can obtain additional locales from DataFlux, a SAS Company, at the following Web address: <http://www.dataflux.com/QKB>. DataFlux regularly adds new locales for various regions and national languages.

If you install additional locales, you need to update your data quality setup file accordingly, as indicated in the documentation that is provided with each locale. Information on locating and editing the setup file is also provided in *SAS Data Quality Server: Reference*.

You can also create new locales (and edit existing ones) using the dfPower Customize software from DataFlux, a SAS Company.

Creating Schemes

A *scheme* is a reusable collection of match codes and standardization values that is applied to input character values for the purposes of transformation or analysis. Schemes can be created in Blue Fusion Data (BFD) format or SAS format (NOBFD). Before your data integration developers can use the Apply Lookup Standardization template in SAS Data Integration Studio, you must specify a scheme repository and a scheme repository type. You also need to create schemes by using the SAS Data Quality Server or dfPower Studio software. For information on how to create schemes, see the relevant product documentation.

Scheme repositories should be separated based on scheme type. BFD schemes and NOBFD schemes should be stored separately to ensure that standardization jobs use the appropriate schemes. Two scheme repositories are provided in the default installation. On the SAS Application Server, the default scheme directory is *path-to-config-dir\Lev1\SASMain\SASEnvironment\QltyKB\scheme*. This directory contains a number of BFD schemes that are supplied with the SAS Data Quality Server software.

A second default scheme directory is provided for interactive SAS sessions that are started on the local host: *sas-root\dquality\sasmisc\content\scheme*.

In SAS Data Integration Studio, the scheme repository and scheme repository type are specified in the **Data Quality** tab of the Options dialog box (select **Tools ► Options**). The default scheme repository type is **dfPower Scheme (BFD)**. No default scheme repository is specified. When you specify a scheme repository in SAS Data Integration Studio, a full or explicit path is recommended. Relative paths must be specified relative to the SAS Application Server.

If you change an existing value in the fields **Scheme Repository Type** or **Scheme Repository**, then you need to replace any existing instances of the Apply Lookup Standardization transformation in any existing jobs. Replacement is required because scheme metadata is added to these jobs when they are run for the first time. To update a job to use a different scheme repository, add a new Apply Lookup Standardization transformation to the job, configure the new transformation, delete the old transformation, and move the new transformation into place.

Setting Data Quality Options for SAS Data Integration Studio

You can set several options related to data quality by using the **Data Quality** tab in the Options dialog box in SAS Data Integration Studio (select **Tools ► Options**). The following table defines the available options.

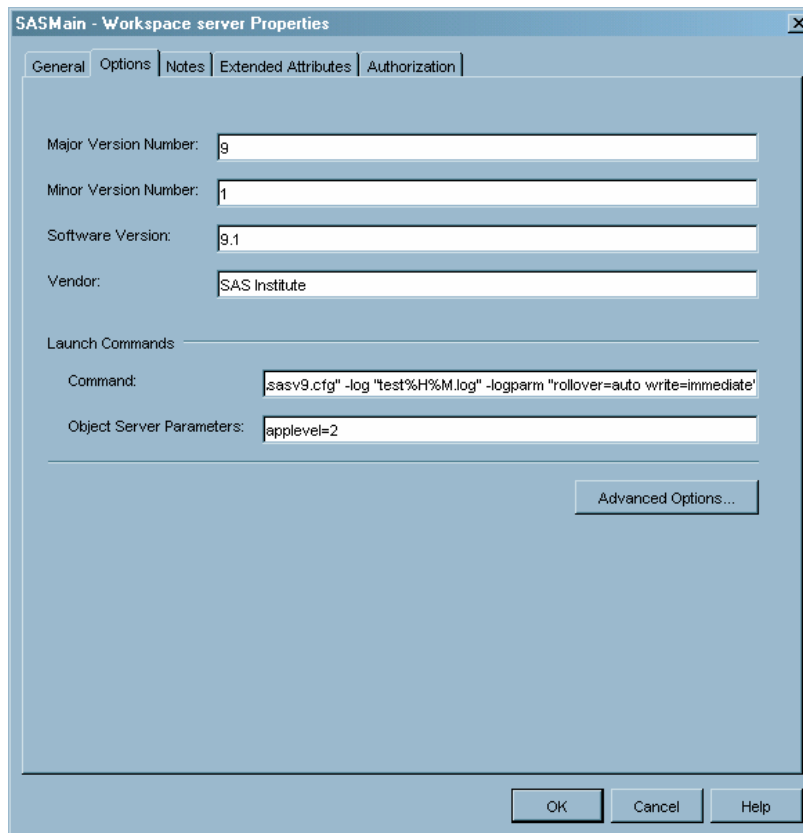
Table 3.3 Data Quality Options in Data Integration Studio

Field	Explanation
Default Locale	By default, this value is set to <Use the default value defined on the server> . Unless you have edited the sasv9.cfg file in your configuration directory, the default locale is ENUSA. You can change the default locale by selecting a different value from the drop-down list.
DQ Setup File Location	By default, this value is also set to <Use the default value defined on the server> . Unless you have edited the sasv9.cfg file in your configuration directory, the setup file dqsetup.txt will be located in the same directory as the sasv9.cfg file. Use the keyboard, drop-down list, or Browse button to specify another setup file.
Scheme Repository Type	By default, the repository type is dfPower scheme (BFD) . Any schemes that you create in SAS format (NOBFD) need to be stored in a separate repository. Use the drop-down list to switch between the two repository types.
Scheme Repository	Specifies the path of the BFD or NOBFD scheme repository. BFD repositories need to be specified as SAS libraries. NOBFD repositories need to be specified with a path. Use the keyboard, drop-down list, or Browse button to specify a different repository. Always store BFD and NOBFD schemes in separate repositories.
Enable launcher	Enables the field DataFlux installation folder , displays the dfPower Profile Launcher icon in the Shortcuts bar, and adds the dfPower Profile Launcher option to the Tools menu.
DataFlux installation folder	Specifies the location of the DataFlux installation directory. The default value is C:\Program Files\DataFlux\dfPower Studio\release-number . Use the keyboard, drop-down list, or the Browse button to specify a different installation folder.

Redirecting Output and Logging Information to a File

SAS Data Integration Studio jobs generate SAS code when they are executed. It is sometimes useful to designate destinations for the outputs and logs that are associated with this code. To specify alternative destinations for the SAS log and SAS output, add the following options to the **sas** command that starts the SAS Workspace Server.

- 1 In SAS Management Console, expand the **Server Manager**; then, expand the **SASMain - Logical Workspace Server**. You will see a tree node that represents the physical SAS Workspace Server.
- 2 Right-click the icon for the physical SAS Workspace Server, and select **Properties** from the pop-up menu. A Workspace Server Properties dialog box appears.
- 3 Click the **Options** tab. You will see the information that is shown in the following display.



- 4 Edit the text in the **Command** text box. By default, this text is set to

```
sas -config "path-to-config-dir\Lev1\SASMain\sasv9.cfg"
```

To route the SAS log to a file, edit the command to make it look something like this:

```
sas -config "path-to-config-dir\Lev1\SASMain\sasv9.cfg"
-log log-file-name%H%M.log -logparm "rollover=auto"
```

For routing the SAS output to a file, the command should look like this:

```
sas -config "path-to-config-dir\Lev1\SASMain\sasv9.cfg"
-print print-file-name.lst
```

- 5 If you are redirecting the log, enter the following parameter in the **Object Server Parameters** text box:

```
applelevel=2
```

This will allow an appropriate level of log information to be routed to the file.

CAUTION:

Do not add the options to the configuration file specified in this command. Setting values in `Lev1\SASMain\sasv9.cfg` affects every server that is launched. This includes the SAS Metadata Server, the SAS OLAP servers, and every SAS Workspace Server and SAS Stored Process Server. \triangle

- 6 Click **OK** in the Workspace Server Properties dialog box.

Enabling Status Code Handling

Overview of Status Code Handling

When a data integration developer executes a job in SAS Data Integration Studio, notification of the job's success or failure can be e-mailed to a person, can be written to a file, or can cause the execution of an autocall macro. A **Status Handling** tab is included in the property windows for jobs and for some transformations. Users can select options from a list of code conditions and actions on this tab. For example, a user can select a code condition such as **successful** and associate it with an action such as Send Email.

Before data integration developers can use some of the actions, you must set up the environment properly. Such setup is required for the following actions:

- E-mail actions—You must set SAS system options for e-mail for the SAS Application sServer that is used to execute jobs.
- Custom actions—You must make a SAS macro autocall library accessible by the SAS Application Server that is used to execute jobs.
- Send Entry to a Data Set—You must pre-assign the library that contains the data set to a SAS Application Server before the job or transformation executes.

For information on how to support the first two actions listed above, see the subsections below. For information on how to support the last action, see Assigning Libraries.

Supporting the E-mail Action

Setting up the e-mail action is simple. Just add the appropriate SAS system options for e-mail to the configuration file *path-to-config-dir\Lev1\SASMain\sasv9.cfg*. For example, if you are using the SMTP e-mail interface, you would add to this file the following options:

```
-emailsys smtp
-emailhost email-server
```

In this case, the value of *email-server* specifies the SMTP server that supports e-mail access for your site.

Note: The e-mail system options are documented in SAS Help and Documentation. △

Supporting the Custom Action

Using the Custom action, a data integration developer can execute a macro in a macro autocall library in order to provide user-defined status handling for a job or transformation. On the administrative side, all you need to do is to make sure that the autocall library is known to the SAS Application Server. You do this by editing the file *path-to-config-dir\Lev1\SASMain\sasv9.cfg*. By default, the file contains a line similar to this:

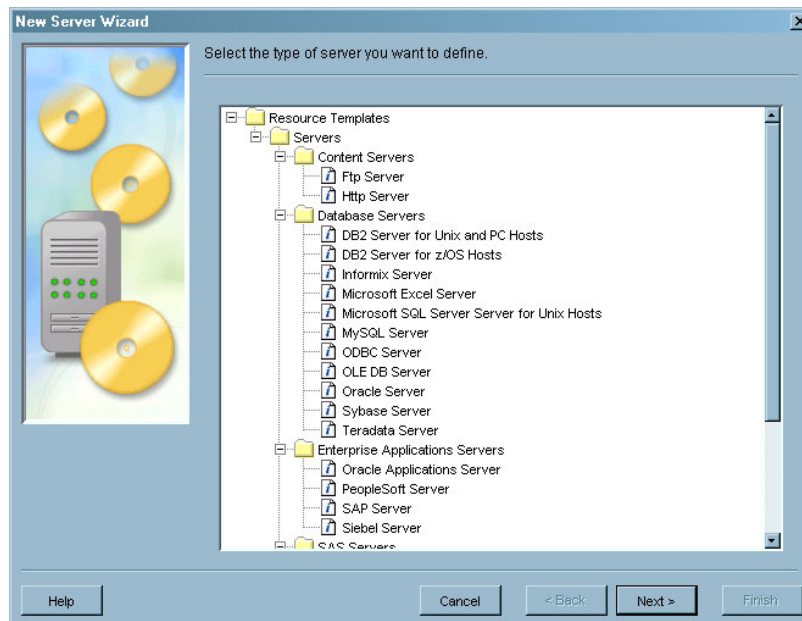
```
-sasautos ("SASEnvironment/sasMacro" SASAUTOS)
```

Add the library to the list of libraries in parentheses. The path to the library can be a full path or a path relative to *path-to-config-dir\Lev1\SASMain*.

Enabling the External File Wizards to Retrieve Files Using FTP or HTTP

External files are often found on FTP or HTTP servers. In order to access these files, you must resolve the physical path to the server. You define an FTP or HTTP server with the Server Manager plug-in to SAS Management Console. To launch the New Server wizard from SAS Management Console, complete these steps:

- 1 Use your metadata profile to log on to the SAS Metadata Server that contains the SAS Metadata Repository connection that you want to use.
- 2 In the SAS Management Console navigation tree, select **Environment Management ► Server Manager**.
- 3 To display the first wizard window, select **Actions ► New Server**.

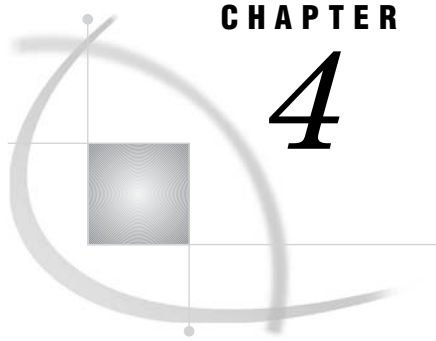


- 4 Select either the FTP or HTTP server type, then click **Next** to continue.
- 5 When defining the server in the New Server wizard, specify these properties:

Server type	Enter FTP or HTTP .
Name	Enter a name.
Software Version	Enter the software version.
Vendor	Enter the name of the vendor.
Application Server Type (HTTP only)	Enter the type of the SAS Application Server.
Base Path(s) (HTTP only)	Enter the full path to the server.
Authentication Domain	Enter your default authentication domain (DefaultAuth).
Application Protocol (HTTP only)	Enter either HTTP or HTTPS .

Host Name	Enter the host name of the machine on which the server runs.
Port Number	Enter 21 (80 for HTTP).
Proxy URL (optional, HTTP only)	Secure Socket Layer (SSL) support is necessary to use a proxy server. You need to download the SAS Secure Software package from <code>http://www.sas.com/apps/demosdownloads/setupintro.jsp?listing=cat=none</code> .

For more information about how to define a server, see the *SAS Management Console: User's Guide*.



CHAPTER

4

Administering SAS Enterprise Guide

<i>Overview of Administering SAS Enterprise Guide</i>	49
<i>Connecting SAS Enterprise Guide to SAS Servers in the SAS Intelligence Platform</i>	49
<i>Reviewing Library Access in SAS Enterprise Guide</i>	50
<i>Enabling ESRI Maps in SAS Enterprise Guide</i>	52

Overview of Administering SAS Enterprise Guide

SAS Enterprise Guide provides a graphical interface into the SAS programming language. Wizards and menu options enable you to quickly generate graphical projects, tasks, and process flows. The process flows analyze data, generate graphs, generate reports, and publish reports. Behind each process flow is a SAS program that you can execute and edit.

As part of the SAS Intelligence Platform, SAS Enterprise Guide is frequently used to query OLAP cubes and generate stored processes.

After you install SAS Enterprise Guide, you need to connect the client to the SAS Metadata Server and possibly other SAS servers. You also need to examine the accessibility of data libraries based on the access requirements of the client. Additional installation and configuration steps are required to use an ESRI server to integrate your SAS OLAP cube data with geographic maps using SAS Enterprise Guide.

To learn more about SAS Enterprise Guide, see the usage information in the online Help that is installed with the software. See also the SAS Enterprise Guide documentation Web site, which provides system requirements, installation instructions, and *Administering SAS Enterprise Guide*, at <http://support.sas.com/documentation/onlinedoc/guide/>. The SAS Enterprise Guide home page is available at http://www.sas.com/technologies/bi/query_reporting/guide/index.html

Connecting SAS Enterprise Guide to SAS Servers in the SAS Intelligence Platform

To connect SAS Enterprise Guide into the SAS Intelligence Platform you need to create a connection to the SAS Metadata Server and to other servers as needed by each client.

Follow these steps to connect a SAS Enterprise Guide client into the SAS Intelligence Platform:

- 1 Start SAS Enterprise Guide.
- 2 In the menu bar, select **Tools** ► **SAS Enterprise Guide Explorer**.

- 3 In the menu bar of SAS Enterprise Guide Explorer, select **File ► Manage Repositories**.
- 4 In the Repository Manager dialog box, click **Add**.
- 5 In the Add Repository Definition dialog box, enter a name and description that identifies the computer that hosts the SAS Metadata Server.
- 6 In the **Machine** field, select **Remote** and enter the network name of the metadata server. The default port is 8561.
- 7 Type the user name and password in the **User** and **Password** fields. Include the authentication domain in the user name if you have separate authentication domains on your network.
- 8 In the **SAS Metadata Repository Name** field, click **Browse** to see the repositories that are available on the server. Select **Foundation** unless you need to use a specific lower-level repository for this user.
- 9 Click **Save** to close the Add Repository Definition dialog box and return to the Repository Manager dialog box.
- 10 In the Repository Manager dialog box, select the new repository definition and click **Set active**. Then close the Repository Manager dialog box.
- 11 In the Enterprise Guide Explorer window, expand **Servers** in the left pane. The default repository name is **SASMain**. Select **SASMain** and authenticate as directed. When the connection is made, browse the libraries and files to confirm their availability.
- 12 Close the SAS Enterprise Guide Explorer window.

Reviewing Library Access in SAS Enterprise Guide

When you create a library using SAS Management Console, the library is defined on the SAS Metadata Server, and the library assignment is shared across SAS servers. In SAS Enterprise Guide, access to registered libraries is determined in part by the method that is used to assign those libraries.

Follow these steps to ensure appropriate library access for each SAS Enterprise Guide client.

- 1 Start SAS Enterprise Guide and select **Tools ► SAS Enterprise Guide Explorer**.
- 2 In the left pane, expand **Servers** and select **SASMain**.
- 3 Display the libraries that are defined for that metadata repository.
- 4 Right-click one of the libraries that will be accessed from that client, and select **Properties**.
- 5 In the Properties window, select **Assignment**.
- 6 Review the library assignment method based on the access requirements of the client and the access controls on the registered libraries. The library assignment methods are defined as follows:

SAS Enterprise Guide

assigns the library based on the engine and LIBNAME statement options that are defined on the SAS Metadata Server. When you choose this library assignment method, SAS Enterprise Guide ignores table and column metadata access control (Read, Write, Create, Delete).

Use this library access method when the client requires Write access to the library, and when the tables in the library are not subject to access control. If Write access to a registered library is not required, use **Metadata Library Engine**.

Write access to registered libraries should be closely monitored because the SAS code that is generated by SAS Enterprise Guide does not automatically synchronize metadata. Changes to columns are written into the physical table, but are not written into the table metadata.

Note that you can edit the code that is generated by SAS Enterprise Guide to include the METALIB procedure, which can be used to update table metadata.

SAS Server

assigns the library by merging the library definition from the current SAS server with the library metadata that is stored on the SAS Metadata Server. Metadata access control for tables and columns is not enforced. The library is assumed to be pre-assigned on the SAS server using an AUTOEXEC or the METAUTOINIT process that runs when the server is initialized.

Use this library access method when you want the library to be assigned consistently across clients, including SAS Enterprise Guide, the SAS Add-In for Microsoft Office, and the SAS Stored Process Server.

Note that when you create a new library using SAS Enterprise Guide and select **SAS Server** as the library assignment method, you do not automatically register that new library on the SAS Metadata Server. Use SAS Management Console to register these libraries.

Metadata Library Engine

assigns the library using the metadata LIBNAME engine. By default (with the checkbox selected), the library is read-only, you cannot add or create new tables, and SAS Enterprise Guide displays registered tables only. Use this library assignment method to protect the referential integrity of the table metadata. This method is recommended for OLAP cube libraries.

To enable Write access at the library level and enforce metadata access control at the table and column level, deselect the check box labeled **Show only tables with metadata definitions**. With this library access method, SAS Enterprise Guide displays all of the tables in the library, including any tables that are not registered.

Note that when you create a new library and specify the **Metadata Library Engine**, the library is automatically registered on the SAS Metadata Server.

- 7 If the assignment method for the selected library does not meet the needs of the SAS Enterprise Guide client, use SAS Management Console to change the library assignment, create new libraries, or change access permissions.
- 8 At a minimum, make sure that you review the assignment methods for all of the registered libraries that require Write access from SAS Enterprise Guide clients.
- 9 It might be necessary to ask SAS Enterprise Guide users to contact you before they change a library assignment method.

Note that the library assignment methods that you select in SAS Enterprise Guide also apply to the SAS Add-In for Microsoft Office. For more information on the SAS Add-In for Microsoft Office, see Chapter 2, “Administering the SAS Add-In for Microsoft Office,” on page 3.

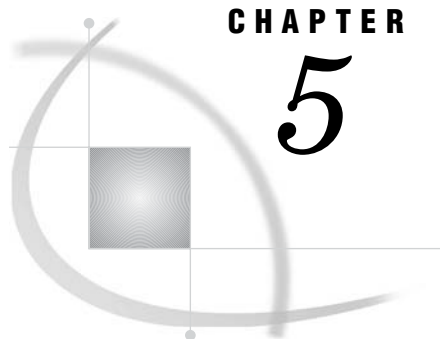
For information on metadata access control, see the section on the metadata authorization layer in the *SAS Intelligence Platform: Security Administration Guide*, at <http://support.sas.com/documentation/onlinedoc>.

For information on pre-assigned libraries, see “Pre-assigning Libraries” in the *SAS Intelligence Platform: Data Administration Guide*.

Enabling ESRI Maps in SAS Enterprise Guide

Digital maps from Environmental Systems Research Institute, Inc. can be integrated into OLAP cube data by installing an ESRI server. With SAS Enterprise Guide, users can query ESRI-enabled OLAP cubes by selecting geographic regions.

To enable the ArcGIS software in SAS Enterprise Guide, you need to install and configure an ESRI server, and purchase and install the ArcGIS Engine Runtime component on each SAS Enterprise Guide client. Complete information on ESRI integration is provided in “Configuring the ESRI Map Component” in the *SAS Intelligence Platform: Web Application Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.



CHAPTER

5

Administering SAS Enterprise Miner

<i>Overview of Preparing SAS Enterprise Miner for Use</i>	53
<i>Configuring SAS Enterprise Miner</i>	54
<i>Overview of Configuring SAS Enterprise Miner</i>	54
<i>Enterprise Miner Macro Variables</i>	55
<i>Enterprise Miner Install Directory</i>	56
<i>Java Virtual Machine Properties</i>	56
<i>The Enterprise Miner Configuration File</i>	56
<i>Java Web Start Properties</i>	56
<i>Customizing SAS Enterprise Miner Metadata Using the Enterprise Miner Plug-in for SAS Management Console</i>	57
<i>Server Extensions</i>	57
<i>Model Management Configuration</i>	58
<i>Other features</i>	58
<i>Getting Started</i>	58
<i>The Projects Folder</i>	58
<i>Customize Servers for Enterprise Miner</i>	59
<i>The Models Folder</i>	60
<i>Specify the WebDAV Server</i>	60
<i>Create Model Groups</i>	60
<i>Customizing the Apache Tomcat HTTP Server</i>	61
<i>Securing SAS Enterprise Miner Metadata</i>	63
<i>Overview of Securing SAS Enterprise Miner Metadata</i>	63
<i>Securing Access at the SAS Enterprise Miner Folder Level</i>	63
<i>Securing Access at the Projects Folder Level</i>	64
<i>Securing Access at the Individual Project Level</i>	65
<i>Securing Access at the SAS Workspace Server Level</i>	66

Overview of Preparing SAS Enterprise Miner for Use

SAS Enterprise Miner is the first and only data mining solution that addresses the entire data mining process. Combined with SAS data warehousing and OLAP technologies, SAS Enterprise Miner helps companies reveal trends, explain known outcomes, predict future outcomes, and identify factors that can produce a desired effect.

There are two ways to deploy SAS Enterprise Miner:

- *Personal workstation.* In this deployment, Java files, configuration files, and documentation are stored locally on the client computer. The client communicates directly with the SAS Workspace Server and the SAS Metadata Server and must remain connected for the duration of a model training event. The personal workstation installation is appropriate for single-user configurations.

- *Enterprise client and Analytics Platform server.* The SAS Enterprise Miner enterprise client uses only the Java files that are needed for display on the client computer. The other components of Enterprise Miner run within an Analytics Platform server, to which Enterprise clients can connect. All configuration is managed by the shared Analytics Platform installation, thus eliminating the need for any client configuration. This installation also facilitates multiple users working on projects collaboratively. Users can work in the same project, disconnect and reconnect to model training processes, and share mining results packages without experiencing resource conflicts.

As a best practice, you can deploy the SAS Enterprise Miner client using Java Web Start, which enables you to deliver enterprise client files on demand from a link in a Web document. Java Web Start automatically downloads the most recent versions of required files from a SAS Application Server and stores them on the client computer. If the Analytics Platform is configured to run its embedded HTTP server, then the Java Web Start feature is enabled automatically.

Note: For more information about how to deploy SAS Enterprise Miner, see the SAS Enterprise Miner Help. \triangle

During a planned SAS installation, a SAS Metadata Server, an object spawner, and a SAS Workspace Server are defined and configured to be available for use by SAS Enterprise Miner. The deployment process also creates a foundation metadata repository and some initial users.

In addition, if this is not a personal workstation installation, you must perform the following tasks:

- create additional SAS Enterprise Miner users
- secure the metadata definitions for projects and models that are created by SAS Enterprise Miner users

Note: You might need to configure the Apache Tomcat HTTP Server for use with the SAS Enterprise Miner model repository. \triangle

Configuring SAS Enterprise Miner

Overview of Configuring SAS Enterprise Miner

Enterprise Miner provides some configuration tools and files that allow you to customize the behavior of the application. All configuration is managed centrally. In the case of a personal desktop installation, this means that all of the configuration is managed on your desktop system. In the case of a multi-user installation, Enterprise Miner configuration is managed on the server hosting the remote Enterprise Miner components via the SAS Analytics Platform server. Clients connecting to an Analytics Platform server do not require individual configuration.

Enterprise Miner configuration can be divided into two parts:

- 1 configuration of Enterprise Miner features
- 2 configuration of the SAS Analytics Platform, which provides the infrastructure for Enterprise Miner, and other analytical applications

Normally, the SAS Analytics Platform will already be configured after installation has completed. However, there may be cases in which it is necessary to reconfigure it using its own configuration wizard. For more information about the SAS Analytics Platform, see the *SAS Analytics Platform Administrator's Guide*. For more information about how

to customize some metadata objects for SAS Enterprise Miner, see “Customizing SAS Enterprise Miner Metadata Using the Enterprise Miner Plug-in for SAS Management Console” on page 57.

Enterprise Miner Macro Variables

Enterprise Miner can be further customized by the use of some SAS macro variables in the Enterprise Miner project startup code.

Table 5.1 Enterprise Miner Macro Variables

Macro Variable	Default Value	Description
EM_DECMETA_MAXLEVELS	32	Controls the maximum number of levels for a target variable for which a target profile can be built.
EM_TRAIN_MAXLEVELS	512	Controls the maximum number of levels for a target variable when building a Data Mining Database (DMDB).
EM_PMML	N	Controls the generation of Predictive Modeling Markup Language (PMML) score code. When this macro variable is set to Y (or YES) the PMML score code is generated by nodes that support it (Association, Clustering, Tree, Regression, Neural, Autoneural).
EM_EXPLOREOBS_MAX		Controls the maximum number of observations downloadable for visualization. When this value is blank, the maximum number of rows downloaded is based on the record length.
EM_EXPLOREOBS_DEFAULT		Controls the default number of observations downloadable for visualization. When this value is blank, the default number of rows downloaded is based on the record length.
EM_DISPLAY_UDF	N	If set to Y, then any user-defined formats found in the format search path are downloaded whenever the project is open, and are subsequently used when displaying data for tables that reference those formats. Otherwise, only raw data values are displayed. Once downloaded, the formats are used for the remainder of your Enterprise Miner session, even though you may later set this macro variable to N. However, if the macro variable is set to Y, the download occurs every time you open the project in order to ensure that the downloaded copies are up to date.

Enterprise Miner Install Directory

The Enterprise Miner install directory is located in the SAS Analytics Platform *applications* directory. If the Analytics Platform is installed at the location **C:\Program Files\SAS\SASAPCore**, then the Analytics Platform applications directory is **C:\Program Files\SAS\SAS\APCore\apps** and the Enterprise Miner install location is **C:\Program Files\SAS\SASAPCore\apps\EnterpriseMiner**.

Java Virtual Machine Properties

A number of properties can be used to customize some aspects of Enterprise Miner behavior. These properties are passed to the Java Virtual Machine as options to the Java command used to launch Enterprise Miner. If you are launching Enterprise Miner using the provided Java Web Start link available from the Analytics Platform status page, you will need to set these properties in the Enterprise Miner configuration file (see the following section). Otherwise, you can edit either the `em.ini` file (on Windows) or the `em` launch script (UNIX) to add the properties to the Java command. Either file can be found in the Enterprise Miner *executables* directory called **bin**, which is an immediate sub-directory of the Enterprise Miner install directory.

Table 5.2 Java Virtual Machine Properties

Property	Default Value	Description
<code>em.maxlines</code>	20,000	Maximum number of lines of text fetched from the SAS server before truncation occurs. This value is used, for example, in the log and output windows of the results viewers to avoid out-of-memory conditions that may otherwise occur.
<code>em.appserver.url</code>	none	The default Analytics Platform host (normally used for Java Web Start deployments — see the following section).

The Enterprise Miner Configuration File

Java Web Start Properties

Located in the install directory is a file called **app.config** that controls some properties of Enterprise Miner for all clients.

You can edit the **app.config** file by using a text editor to customize the following properties.

These properties control how clients are launched from the Analytics Platform status page using Java Web Start. The embedded HTTP server for the Analytics Platform must be enabled in order to provide Java Web Start as a deployment option.

Table 5.3 Java Web Start Properties

Property	Description
webstart.client.initial.heap.size	Controls the initial amount of heap memory allocated to clients
webstart.client.max.heap.size	Controls the maximum amount of heap memory allocated to clients
webstart.client.vm.properties	Initial properties that can be passed to the Java Virtual Machine (JVM) when clients are launched. The properties are specified using the following syntax: <i>-Dproperty-name=property-value</i> For example: <code>-Dem.maxlines=10000</code>

By default the property `em.appserver.url` is set as follows:

```
-Dem.appserver.url=rmi://$RMI_HOST$: $RMI_PORT$
```

Whenever a request is made from a web browser to launch Enterprise Miner using Java Web Start, this property is updated with the host name and port of the Analytics Platform server serving the request, to form a valid RMI-based URL. For example, if the Analytics Platform server is called `myhost.mydomain.com` and the RMI registry port is 5099, then the server resolves this property as follows:

```
-Dem.appserver.url=rmi://myhost.mydomain.com:5099
```

`RMI_HOST` and `RMI_PORT` are replaced with the actual host name and port. This property informs the Enterprise Miner client of the location of the RMI registry to contact in order to establish a link with the Analytics Platform server.

Customizing SAS Enterprise Miner Metadata Using the Enterprise Miner Plug-in for SAS Management Console

Server Extensions

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

- set the default locations for new projects on a particular server, and specify whether these locations can be modified by users when creating new projects.
- set the maximum number of concurrent tasks allowed on a particular server. This setting affects the extent of parallel processing applied to sibling branches running in a data mining process flow, so that you can tune your SMP environment for optimal performance.
- specify SAS initialization code to be run whenever a project is opened, or whenever a process flow is run to generate diagram-level results or exportable model results packages. This is similar to the Project Start-up Code feature of Enterprise Miner

projects, except that it is specified on a server so that you can initialize all SAS sessions created for all projects associated with a particular server.

- provide an alternate command to launch the SAS/CONNECT sessions that are used for each node running in a data mining process flow.

Model Management Configuration

The Enterprise Miner plug-in also lets you configure an optional WebDAV server to act as the model repository storage center, so that mining results packages registered to the model repository from your Enterprise Miner projects are copied to a WebDAV server location. These packages can then be accessed by the model consumers throughout your organization.

You can also use the plug-in to create groups to help you categorize your various classes of models.

Note: The configuration features for model management provided by this plug-in replace the functionality found in the Enterprise Miner configuration wizard that was bundled with earlier releases of this product. Δ

Other features

The Enterprise Miner plug-in provides some basic browsing functionality to help you navigate through the list of projects and mining result packages registered to the Foundation repository.

Getting Started

In order to use the Enterprise Miner plug-in, you must have first started an Enterprise Miner client at least once, so that the metadata repository can be initialized with Enterprise Miner metadata. Once you have successfully logged on to Enterprise Miner, you can log off and start SAS Management Console. It is not necessary to have defined or opened any Enterprise Miner projects at this stage.

The plug-in installs into your locally installed SAS Management Console directory when you install Enterprise Miner. Once it is installed, you will see the Enterprise Miner application icon appear under the Application Management section in the left pane of the Management Console window.

Expand **Enterprise Miner** to display the **Projects** and **Models** folders, as shown in the following display:



The Projects Folder

This folder contains a list of all Enterprise Miner projects registered in the Enterprise Miner repository. Click on the folder to see the list of all projects in the right

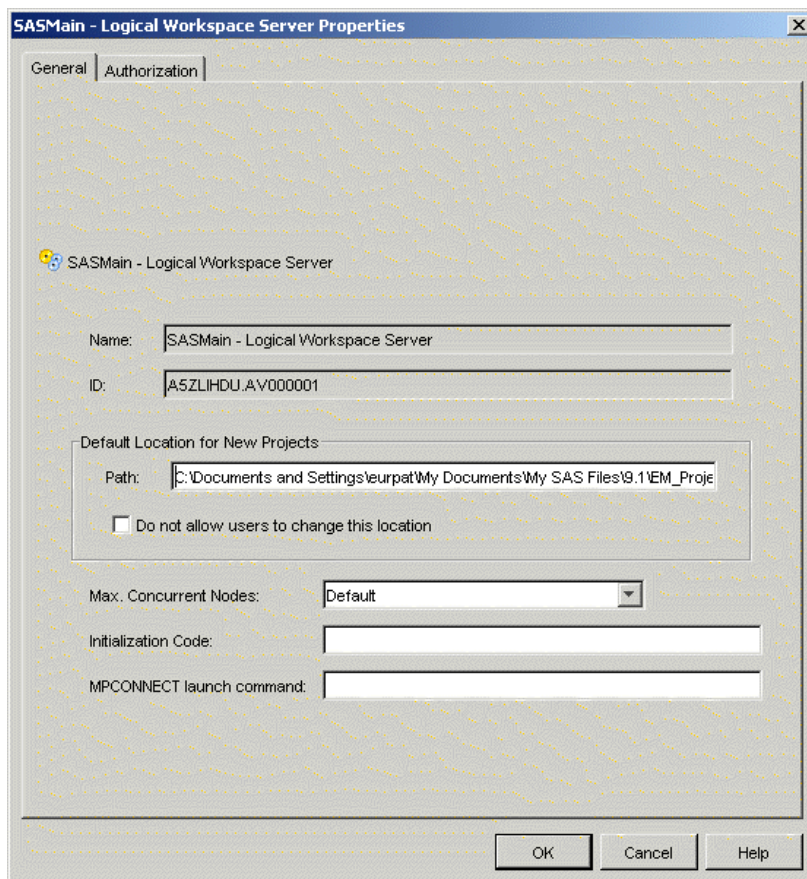
console pane. By expanding the Projects folder, you will find the projects organized by Logical Workspace Server. Clicking on a Logical Workspace Server icon in the left pane will subset the list of projects on the right, so that only those projects belonging to that server are displayed.

You do not need to have any registered projects in order to view the Logical Workspace Servers, or to customize them for Enterprise Miner. Servers for which no projects exist will have their customizations recognized whenever new projects are created on those servers.

Customize Servers for Enterprise Miner

To customize the servers for Enterprise Miner, perform the following steps:

- 1 Right-click a server icon under the **Projects** folder, and select **Properties** from the pop-up menu.



- 2 In the **Path** box, enter a default location for new Enterprise Miner projects created on this server.
- 3 Select **Do not allow users to change this location** if you want to prevent the user who is creating the project from changing the location.
- 4 From the **Max. Concurrent Nodes** drop-down list, select the maximum number of concurrently running branches in process flows.
- 5 In the **Initialization Code** box, enter the path (on this server) to a text file containing SAS code that you would like to run any time a project is opened, or a run or report action is performed on a process flow.

- 6 In the **MPCONNECT launch command** box, enter an alternate command to use for launching MPCONNECT sessions. Normally it is safe to leave this box blank, but there may be cases where you would like to modify some SAS system defaults for sessions that are used when running process flows. The following default command is used when this box is left blank:

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

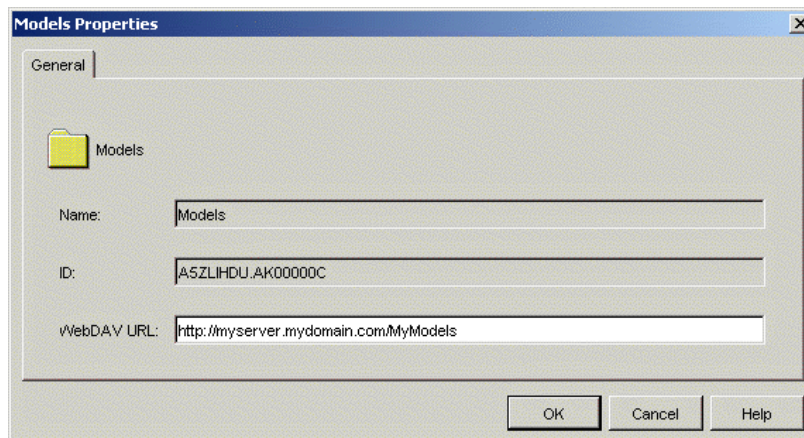
This command has the same effect as using the SAS command that was used to launch the SAS workspace session at the time the project was opened.

The Models Folder

Specify the WebDAV Server

You can use the Models folder to specify a URL to your WebDAV server. If specified, model result packages will be copied to this location at the time they are registered from the Enterprise Miner project environment. To specify the WebDAV server, perform these steps:

- 1 Right-click the **Models** folder and select **Properties** from the pop-up menu.



- 2 In the **WebDAV URL** box, enter the URL to your Web server.

Create Model Groups

To create model groups, perform the following steps:

- 1 Right-click the **Models** folder and select **New Group** from the pop-up menu.
- 2 Enter a new group name at the prompt. There is currently no support for sub-groups, and duplicate group names are not allowed.

Customizing the Apache Tomcat HTTP Server

The HTTP server recommended for use with SAS Enterprise Miner 5.2 is the Apache Tomcat 4.1.18 server with integrated WebDAV support. To enable the model registration and storage functions of SAS Enterprise Miner, you must enable the Tomcat WebDAV write functionality and define the context path for the Enterprise Miner Model Viewer application for Tomcat.

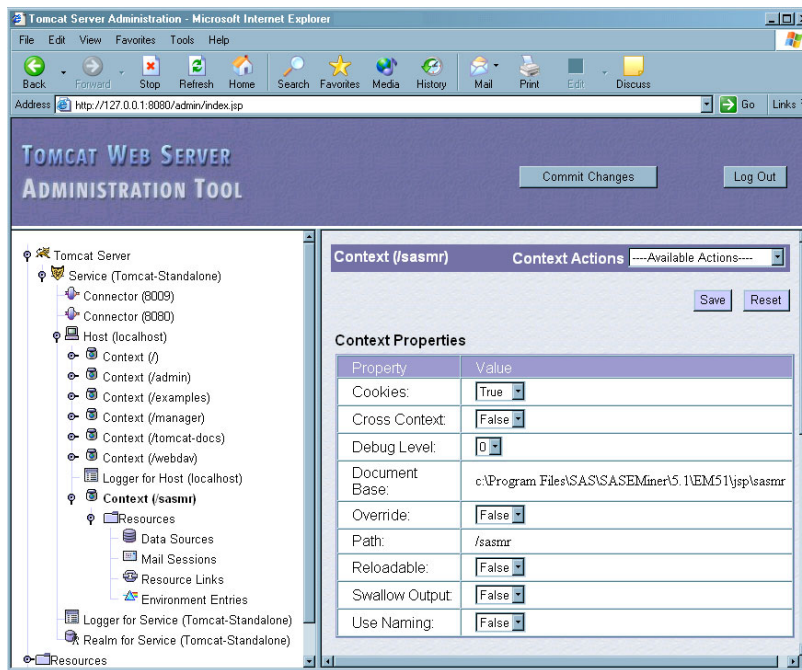
Note: Servers other than Tomcat 4.1.18 might have different instructions. Δ

- 1 To enable the Tomcat write functions, remove the comment delineators from the following block of code in the WebDAV configuration XML file. The file is typically found in

```
c:\Program Files\Apache Group\Tomcat  
4.1\webapps\webdav\WEB-INF\web.xml.
```

```
<!--  
  <init-param>  
    <param-name>readonly</param-name>  
    <param-value>>false</param-value>  
  </init-param>  
-->
```

- 2 As a best practice, use the Tomcat Web Server Administration Tool to define the SAS Enterprise Miner Model Viewer to Tomcat. On Windows, you can launch the tool by selecting **Start ► Programs ► Apache Tomcat 4.1 ► Tomcat Administration**.
 - a In the navigation tree on the left, select **Tomcat Server ► Service ► Host**.
 - b In the display area to the right of the navigation tree, select **Create New Context** from the **Host Actions** list.
 - c In the **Document Base** field, enter the path to the viewer. Typically, the path is
C:\Program Files\SAS\SASAPCore\apps\EnterpriseMiner\webapps\sasmr
 - d In the **Path** field, enter the context path. Typically, the path is **/sasmr**.
 - e Click **Save**. The context is added to the selected Host.



If you do not want to use the Tomcat Web Server Administration Tool, you can manually add the following context coding beneath the `<Host ...>` tag in the Tomcat server configuration XML file. The file is typically found in

C:\Program Files\Apache Tomcat 4.1\conf\server.xml.

```

<!-- EM Model Viewer -->
  <Context path="/sasmr"
    docBase="C:\Program Files\SAS\SASAPCore\apps\EnterpriseMiner\webapps\sasmr"
    crossContext="false"
    debug="0"
    reloadable="false">
  </Context>

```

Note: If your SAS Enterprise Miner installation path varies from the one shown above, change the location of the Document Base (**docBase= value**). \triangle

If the SAS Enterprise Miner Model Viewer is installed on a Tomcat server on which Tomcat security is enabled, the installer will need to grant permissions to Catalina for the model viewer. To accomplish this, you need to add the following item to the Catalina permissions file, typically located at

C:\Program Files\Apache Group\Tomcat 4.1\conf\catalina.policy or

C:\Tomcat 4.1\conf\catalina.policy. Append these lines to the end of the module:

```

grant codeBase
  "file:C:/Program Files/SAS/SASAPCore/apps/EnterpriseMiner/webapps/sasmr/-"
{
  permission java.security.AllPermission;
};
grant {
  permission java.security.AllPermission "com.sas.*";
};

```

where **C:/Program Files/SAS/SASAPCore/apps/EnterpriseMiner/webapps/sasmr/** (or its UNIX counterpart) is the docBase path to the **sasmr** application on the local disk.

Note: Use forward slashes even for Windows installations. △

- 3 Stop and restart the Tomcat HTTP server to apply the changes.

Note: To install the link into your Start Menu, you need to access the server administration tool at **http://localhost:8080/admin**. △

Securing SAS Enterprise Miner Metadata

Overview of Securing SAS Enterprise Miner Metadata

For thin-client installations, to secure access to the metadata objects that represent SAS Enterprise Miner projects, you grant or deny access to individual users or groups by using the **Authorization** tab for the following:

- the **SAS Enterprise Miner** folder
- the **Projects** folder
- individual projects
- the SAS Application Server that contains the SAS Logical Workspace Server that is associated with the SAS Enterprise Miner projects

Note: For stand-alone installations, there is no need to secure the metadata. △

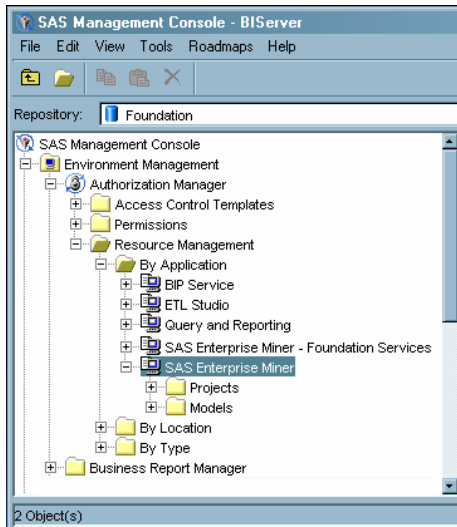
Note: Because all SAS Intelligence Platform applications use the SAS Metadata Server when accessing resources, permissions that are enforced by the SAS Metadata Server provide the strongest protections that are available in the metadata authorization layer. For more information, see “Using the Metadata Authorization Layer” in the *SAS Intelligence Platform: Security Administration Guide*. △

Note: Currently, you cannot secure SAS Enterprise Miner models. △

Securing Access at the SAS Enterprise Miner Folder Level

To access the **Authorization** tab for the **SAS Enterprise Miner** folder in the SAS Management Console navigation tree, complete these steps:

- 1 Use your metadata profile to log on to the SAS Metadata Server that contains the SAS Metadata Repository connection that you want to use.
- 2 Select **Environment Management ► Authorization Manager ► Resource Management ► By Application ► SAS Enterprise Miner**.
- 3 Select the **SAS Enterprise Miner** folder.



- 4 Select **File ► Properties**.
- 5 Click the **Authorization** tab.

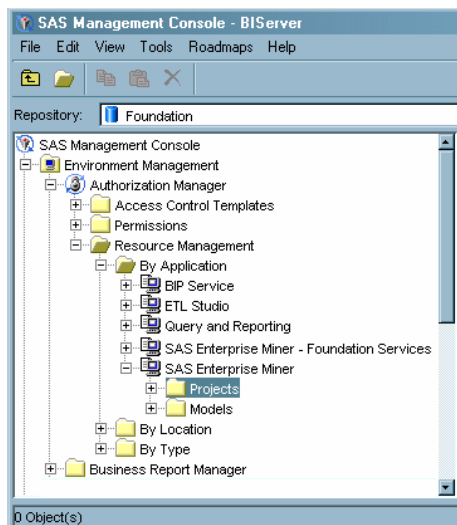
Note: For help on using the **Authorization** tab, click **Help**. △

For example, if you deny ReadMetadata access to the **SAS Enterprise Miner** folder to User A, then User A will not be able to see any projects in SAS Enterprise Miner, unless you explicitly grant ReadMetadata permission to UserA on the Projects folder.

Securing Access at the Projects Folder Level

To access the **Authorization** tab for the **Projects** folder in the SAS Management Console navigation tree, complete these steps:

- 1 Use your metadata profile to log on to the SAS Metadata Server that contains the SAS Metadata Repository connection that you want to use.
- 2 Select **Environment Management ► Authorization Manager ► Resource Management ► By Application ► SAS Enterprise Miner**.
- 3 Select the **Projects** folder.



- 4 Select **File ► Properties**.
- 5 Click the **Authorization** tab.

Note: For help on using the **Authorization** tab, click **Help**. △

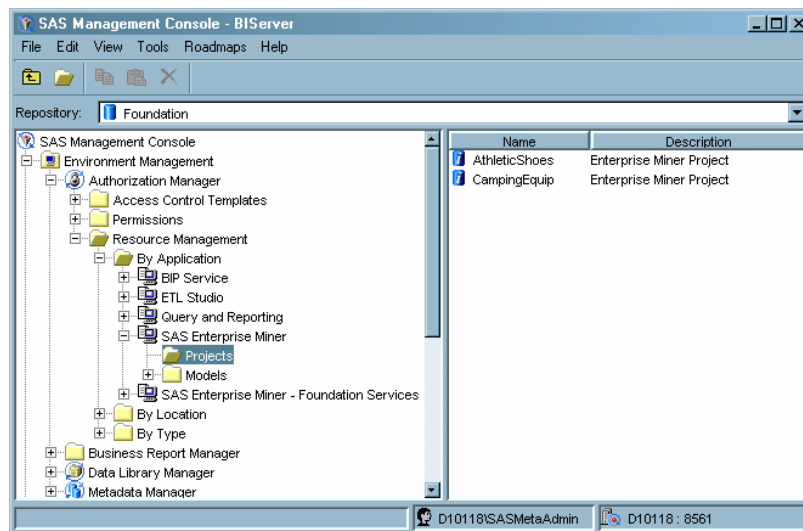
For example, if you deny ReadMetadata access to the **Projects** folder to User A, then User A will not be able to see any projects in SAS Enterprise Miner.

Permissions that you explicitly set on the **Projects** folder will override permissions set on the **SAS Enterprise Miner** folder.

Securing Access at the Individual Project Level

To access the **Authorization** tab for an individual project in the SAS Management Console navigation tree, complete these steps:

- 1 Use your metadata profile to log on to the SAS Metadata Server that contains the SAS Metadata Repository connection that you want to use.
- 2 Select **Environment Management ► Authorization Manager ► Resource Management ► By Application ► SAS Enterprise Miner**.
- 3 Select the **Projects** folder.
- 4 In the display area to the right of the navigation tree, select the project that you want to secure.



- 5 Select **File ► Properties**.
- 6 Click the **Authorization** tab.

Note: For help on using the **Authorization** tab, click **Help**. △

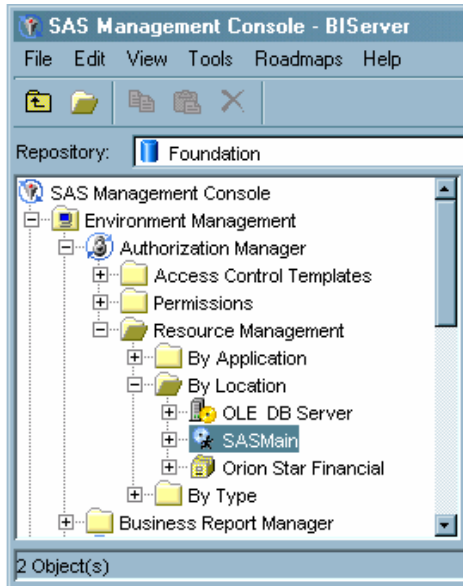
For example, if you deny ReadMetadata access to the **CampingEquip** project to User A, then User A will not be able to see the **CampingEquip** project in SAS Enterprise Miner. If you grant ReadMetadata but deny WriteMetadata, then User A will be able to open the project but will not be able to delete it.

Note: An explicit grant of ReadMetadata permission on the individual project does not override an inherited denial of ReadMetadata from the **Projects** or the **SAS Enterprise Miner** folders. △

Securing Access at the SAS Workspace Server Level

To access the **Authorization** tab for a SAS Application Server in the SAS Management Console navigation tree, complete these steps:

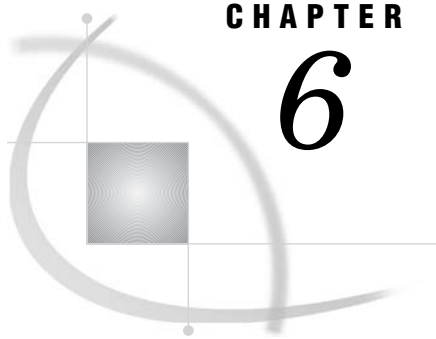
- 1 Use your metadata profile to log on to the SAS Metadata Server that contains the SAS Metadata Repository connection that you want to use.
- 2 Select **Environment Management** \blacktriangleright **Authorization Manager** \blacktriangleright **Resource Management** \blacktriangleright **By Location**.
- 3 Select the SAS Application Server that contains the SAS Workspace Server that is associated with your SAS Enterprise Miner projects.



- 4 Select **File** \blacktriangleright **Properties**.
- 5 Click the **Authorization** tab.

Note: For help on using the **Authorization** tab, click **Help**. Δ

For example, if you deny ReadMetadata access to the **SASMain** application server to User A, then User A will not have access to any projects associated with that server.



CHAPTER

6

Administering SAS Information Map Studio

<i>Overview of Administering SAS Information Map Studio</i>	67
<i>Managing Storage for Information Maps</i>	68
<i>Adding Content for Use by SAS Information Map Creators</i>	68
<i>Input Sources for SAS Information Map Studio</i>	68
<i>Making Data Sources Available to SAS Information Map Studio</i>	69
<i>Making User-Defined Formats Available to SAS Information Map Studio</i>	69
<i>Importing and Exporting Information Maps</i>	69
<i>Considerations for Setting Up Access Control for Information Maps</i>	69
<i>Setting Up Users for SAS Information Map Studio</i>	70
<i>Understanding the Configuration of SAS Information Map Studio</i>	70
<i>Managing the Log File in SAS Information Map Studio</i>	71
<i>Managing the Size of the Java Heap</i>	71

Overview of Administering SAS Information Map Studio

Information maps are metadata objects that represent categories of information from across your enterprise, including data that is dynamically queried or filtered from sources such as SAS OLAP cubes and relational databases. After they are defined, information maps provide a consistent and current data collection that forms the basis for reports generated with SAS Web Report Studio or in stored processes that are created with SAS Enterprise Guide.

SAS Information Map Studio creates, tests, and secures information maps.

The following table outlines the administrative tasks for SAS Information Map Studio.

Table 6.1 Administrative Tasks for SAS Information Map Studio

Administrative Task	Purpose of Task
Managing storage of information maps	Simplifies the process of managing access controls for information maps and ensures that information maps are available to SAS Web Report Studio.
Adding content for use by information map creators	Makes tables, SAS OLAP cubes, user-defined processes, and imported information maps available to users of SAS Information Map Studio.
Controlling access to information maps	Restricts access to information maps in accordance with your security goals.

Administrative Task	Purpose of Task
Setting up users for SAS Information Map Studio	Enables users to log on to SAS Information Map Studio, and ensures that each client is using appropriate encryption.
Understanding your SAS Information Map Studio configuration	Describes for diagnostic purposes the files that affect the operation of SAS Information Map Studio.
Managing the log file in SAS Information Map Studio	Enables the resolution of errors and performance issues.
Managing the size of the Java heap	Improves performance for information maps that involve large amounts of data.

Managing Storage for Information Maps

Proper storage of information maps is important for these reasons:

- If you organize storage of information maps appropriately for your environment, then the management of access controls will be easier.
- If you do not store an information map in an appropriate location within the repository, then that information map will not be available as a data source for SAS Web Report Studio.
- If you change the storage location of an information map, then any reports that were created using the information map in its original location will fail.

To ensure that information maps can be accessed from SAS Web Report Studio, the report storage structure that is registered on the SAS Metadata Server must be identical to the report storage structure that is defined on the WebDAV content server. By default, identical structures are created when you install SAS Web Report Studio. For information maps, the top-level report storage locations must agree. The default name of the top-level storage location is **ReportStudio**.

Information maps can be stored anywhere in the metadata repository. But in order for SAS Web Report Studio to access information maps, they must be stored in a specified location configured as part of your Web Report Studio deployment. The default name of the information map storage location is **ReportStudio\Maps**.

To control access to sets of information maps, create subfolders beneath the **Maps** folder. For example, you might add a subfolder for Human Resources information maps. You can then give a Human Resources user group exclusive access to the Human Resources information map folder. Those access controls will be inherited by all of the information maps within that folder.

To learn more about the report storage structure, see “Setting Up Storage for Reporting” in the *SAS Intelligence Platform: Web Application Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

Adding Content for Use by SAS Information Map Creators

Input Sources for SAS Information Map Studio

The resources that can be used as inputs to SAS Information Map Studio are data sources, stored processes, and imported information maps. SAS Information Map

Studio uses the SAS Metadata Server to access these resources, so they must be registered in the metadata repository.

Making Data Sources Available to SAS Information Map Studio

Each information map must have at least one data source. The data sources for an information map can consist of *either* of these things:

- one or more tables that are registered in the metadata repository and are associated with a library definition. The tables can be any combination of SAS data sets and third-party relational database tables.
- one SAS OLAP cube that is registered in the metadata repository and is associated with a SAS OLAP schema definition.

Users of SAS Information Map Studio must have appropriate permission to access the data sources that they will include in their information maps. For example, in the metadata authorization layer, the following requirements apply:

- In order to include a particular data source in an information map, a user must have ReadMetadata permission for that data source.
- In order to test a query against a SAS OLAP data source, a user must have both ReadMetadata and Read permissions for that data source.
- In order to test a query against a table data source that is accessed with the metadata LIBNAME engine, a user must also have Read permission for the data source.

For information about registering data sources in the metadata repository, see “Connecting to Common Data Sources” in the *SAS Intelligence Platform: Data Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

For information on setting up access control for information maps, see “Considerations for Setting Up Access Control for Information Maps” on page 69.

Making User-Defined Formats Available to SAS Information Map Studio

You must import or register any user-defined formats that have been created for a data set in order to view the data. For more information, see “Establishing Connectivity to a Library of SAS Data Sets” in the *SAS Intelligence Platform: Data Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

Importing and Exporting Information Maps

You can export and import information maps between metadata repositories using SAS Information Map Studio or BI Manager in SAS Management Console. For additional information about using SAS Information Map Studio, see the Help in that software. For additional information about using BI Manager, see “Promoting Individual BI Objects” in the *SAS Intelligence Platform: System Administration Guide*.

Considerations for Setting Up Access Control for Information Maps

Use the metadata authorization layer on the SAS Metadata Server to control access to information maps. To configure the metadata authorization layer with SAS

Information Map Studio, select **Tools ► Authorization**. To use BI Manager in SAS Management Console, select **Environment Management ► BI Manager**.

The relevant permissions for information maps are Read, ReadMetadata, and WriteMetadata. Read access is required to read the data that is referenced in information maps. ReadMetadata access is required to see information maps. WriteMetadata access is required to modify information maps.

The ability to view or work with an information map can be affected by access to each of the information map's underlying components. For information about the metadata-layer requirements for working with information maps and information map folders, see “Access Requirements for Information Maps” in the *SAS Intelligence Platform: Security Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

By default, information maps inherit permissions from their folders. The folders in turn inherit permissions from their parent folders. Additional permissions can be applied as needed to individual information maps within a given folder. For more information, see “Where Can Permissions Be Set?” in the chapter “Understanding Authorization” in the *SAS Intelligence Platform: Security Administration Guide*.

You can apply row-level permissions when you subset data with a prefilter, as described in “BI Row-Level Permissions” in the *SAS Intelligence Platform: Security Administration Guide*.

Setting Up Users for SAS Information Map Studio

The process for adding users to a deployment is documented in “How to Add a User” in the *SAS Intelligence Platform: Security Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>. These aspects of setting up users are specific to SAS Information Map Studio:

- Users must have a metadata identity other than PUBLIC to log on to SAS Information Map Studio. For more information, see “Authentication Overview” in the *SAS Intelligence Platform: Security Administration Guide*, at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>.
- SAS Information Map Studio can cache and reuse the credentials that are supplied by users. This can reduce (or in some cases eliminate) the need to store passwords in the metadata repository. For more information, see “Reuse of Credentials That are Cached from an Interactive Log” in the chapter “Understanding Authentication in the SAS Intelligence Platform” in the *SAS Intelligence Platform: Security Administration Guide*.
- Every time you install SAS Information Map Studio, you must implement encryption for that client in accordance with the encryption strategy at your site. For instructions, see “Enabling Encryption” in the chapter “Securing a Deployment” in the *SAS Intelligence Platform: Security Administration Guide*.

Understanding the Configuration of SAS Information Map Studio

The following administrative and configuration files for SAS Information Map Studio are located on each client:

mapstudio.ini

contains configuration settings for SAS Information Map Studio. For example, you can change the amount of available memory by adding or modifying arguments in this file. In general, you should modify this file only under the direction of SAS

Technical Support. The default location for this file is the installation directory for SAS Information Map Studio, **C:\Program Files\SAS\SASInformationMapStudio\3.1**. For information on changing the size of the Java heap by editing this file, see “Managing the Log File in SAS Information Map Studio” on page 71.

IMS_Log.log

logs for diagnostic purposes selected events that occur during the operation of SAS Information Map Studio. This file name is suggested for use in the Diagnostic Settings dialog box, using the path **C:\Documents and Settings\userid\IMS_Log.log**. By default, the severity level is Error. The default location of the error log is the user’s Profiles directory in Windows. For further information on logging, see “Managing the Log File in SAS Information Map Studio” on page 71.

Managing the Log File in SAS Information Map Studio

The log file in SAS Information Map Studio records selected diagnostic information. To change the information that is recorded, or to change the name or location of the log file, follow these steps:

- 1 Start SAS Information Map Studio.
- 2 Select a metadata profile.
- 3 In the Log On dialog box, click **Settings** to display the Diagnostic Settings dialog box.
- 4 Change log settings and click **OK** to begin using the new diagnostic settings.

By default, the log records at the Error level of severity, in a file that is located under the user’s name in the Windows Profiles directory.

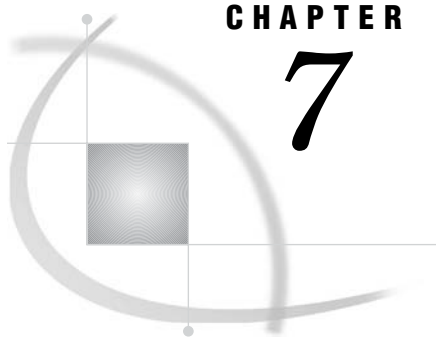
Managing the Size of the Java Heap

To improve performance with very large data sources, follow these steps to increase the size of your Java heap. The default heap size is 1024MB.

- 1 In a text editor, open the initialization file for SAS Information Map Studio:

```
C:\Program Files\SAS\SASInformationMapStudio\3.1\mapstudio.ini
```

- 2 In the initialization file, locate the text string that begins with **CommandLineArgs**. In that text string, locate the arguments **Xms** (minimum heap size) and **Xmx** (maximum heap size). The default values are **-Xms50m -Xmx1024m**, where **m** indicates megabytes.
- 3 Change the heap sizes limits. You can use **g** to indicate gigabytes.
- 4 Save and close the initialization file.
- 5 Start or restart SAS Information Map Studio.



CHAPTER

7

Administering SAS Management Console

SAS Management Console Overview for Administrators 73

Quick Reference for SAS Management Console 73

Understanding the Configuration of SAS Management Console 75

SAS Management Console Overview for Administrators

SAS Management Console is the primary administrative interface for the SAS Intelligence Platform. Use it to manage libraries, security, servers, jobs, metadata, information maps, stored processes, and licenses.

Because of its powerful capabilities, you should limit the number of clients that receive SAS Management Console and grant administrative permissions only to trusted and experienced individuals.

Note: To see the Help for a given component in SAS Management Console, select the component in the tree view, and then open the Help menu in the toolbar. Δ

Quick Reference for SAS Management Console

The following table lists tasks and related components in SAS Management Console:

Table 7.1 Quick Reference for SAS Management Console

Task	Component	Description
Security: manage permissions for repositories, users, and groups	Authorization Manager	See the <i>SAS Intelligence Platform: Security Administration Guide</i> .
Promotion: import and export using packages of objects	BI Manager	Moves entire jobs and information maps.
Reports: create and delete jobs for scheduling, manage content	BI Manager	Works with reports from SAS Web Report Studio.
Stored processes: register, manage, configure	BI Manager Stored Process Manager	Works with stored processes created in SAS Enterprise Guide.

Task	Component	Description
Libraries: define, manage	Data Library Manager	See the <i>SAS Intelligence Platform: Data Administration Guide</i> . See also the client-specific administration information.
Tables: import multiple	Data Library Manager	
Database schemas: define, manage	Data Library Manager	
Geographic maps: define, import services	Map Service Manager	Works with ArcGIS software to embed OLAP data in geographic displays. See “Configuring the ESRI Map Component” in the <i>SAS Intelligence Platform: Web Application Administration Guide</i> .
SAS Metadata Server: control, maintain	Metadata Manager	See the <i>SAS Intelligence Platform: System Administration Guide</i> .
Metadata: promote and replicate repositories and jobs, manage resource templates	Metadata Manager	
Scheduling: create and manage flows	Schedule Manager	Executes job with available dependencies. See the <i>SAS Intelligence Platform: System Administration Guide</i> .
Servers, spawners, external data connections: define, control, manage	Server Manager SAS OLAP Server Monitor	See the <i>SAS Intelligence Platform: Application Server Administration Guide</i> .
Users, groups, and roles: create and manage definitions	User Manager	See the <i>SAS Intelligence Platform: Security Administration Guide</i> .
XML maps: create, import, manage	XML Map Manager	Works with maps from SAS Information Map Studio.
Publishing: manage e-mail accounts, channels, subscribers, message queues	Publishing Framework	Works with SAS Integration Technologies. See the <i>Integration Technologies Administrator’s Guide</i> , at http://support.sas.com/rnd/itech/library/library9.html .

Task	Component	Description
Applications and services: create, deploy, administer	Foundation Services Manager Application Monitor Application Management	Works with SAS Foundation Services and SAS Integration Technologies.
Licenses: display properties	License Manager	Works with site-level license and installation metadata.

Understanding the Configuration of SAS Management Console

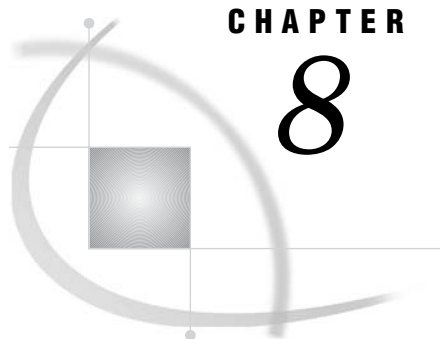
The following administrative and configuration files for SAS Management Console are located on each client:

sasmc.ini

contains configuration settings for SAS Management Console. The default storage location is **C:\Program Files\SAS\SASManagementConsole\9.x**.

app.smc

contains the user profile for SAS Management Console. The default storage location is **C:\Documents and Settings\user-id**.



CHAPTER

8

Administering SAS OLAP Cube Studio

SAS OLAP Cube Studio Overview for Administrators 77
Setting Up the SAS Intelligence Platform for SAS OLAP Cube Studio 78
Understanding the Configuration of SAS OLAP Cube Studio 78

SAS OLAP Cube Studio Overview for Administrators

Use SAS OLAP Cube Studio to build, edit, and tune OLAP cubes. The cube data is physically located on SAS OLAP Servers. End-users query the OLAP cube data by using client cube viewers such as SAS Web Report Studio, SAS Web OLAP Viewer, and SAS Enterprise Guide. Cube viewers submit queries to SAS OLAP Servers in the industry-standard MDX language. The servers respond to queries by delivering result sets to the cube viewers for display. Information on the MDX query language is provided in the *SAS OLAP Server: MDX Guide*.

Note: The documents that are referenced in this section are available at <http://support.sas.com/documentation/onlinedoc/sas9doc.html>. Δ

When you build or edit a cube with SAS OLAP Cube Studio, the application generates a SAS program that is executed on a SAS Workspace Server, which builds the cube on a SAS OLAP Server and creates cube metadata in a specified repository on the SAS Metadata Server. The SAS program that is generated by SAS OLAP Cube Studio contains an OLAP procedure. For information on PROC OLAP, along with other information including error message resolution, see *SAS OLAP Server: User's Guide* and the online Help for SAS OLAP Cube Studio.

SAS OLAP Cube Studio consists of the Create OLAP Schema wizard and the following other components:

- Cube Designer
builds and edits cubes using a wizard format.
- Source Designer
registers source data tables on the SAS Metadata Server.
- Calculated Members
creates members that are calculated when you build the cube, as opposed to loading members from source tables.
- Aggregation Tuning
analyzes queries and suggests aggregations that improve performance.
- Manual Tuning
captures user-defined aggregations and enables the addition of generated aggregations.

Edit Cube Structure

rebuilds existing cubes with different dimensions, hierarchies, levels, members, and aggregations.

Create OLAP Schema

creates new OLAP schemas, which specify the set of SAS OLAP Servers that have access to a set of OLAP cubes. Generally speaking, a new SAS OLAP Server requires a new OLAP schema. For further information on OLAP schemas, see “About OLAP Schemas,” in the *SAS Intelligence Platform: Application Server Administration Guide*.

Setting Up the SAS Intelligence Platform for SAS OLAP Cube Studio

Set up the SAS Intelligence Platform as follows to use SAS OLAP Cube Studio:

- 1 Install, configure, and start a host that runs both a SAS Workspace Server and a SAS OLAP Server. This step generally occurs at initial deployment. See the *SAS Intelligence Platform: Application Server Administration Guide* for details.
- 2 Define libraries and data sources. See the *SAS Intelligence Platform: Data Administration Guide*.
- 3 Define access controls for libraries, users of SAS OLAP Cube Studio, and users of cube viewers. After you build a cube, you can apply access controls within the cube, including member-level access controls, as described in the *SAS Intelligence Platform: Security Administration Guide*.

Understanding the Configuration of SAS OLAP Cube Studio

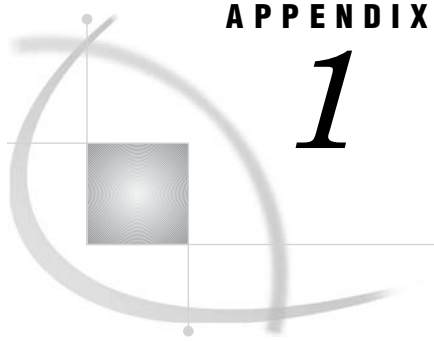
The following administrative and configuration files for SAS OLAP Cube Studio are located on each client:

olapa.ini

contains configuration settings for SAS OLAP Cube Studio. The default storage location is **C:\Program Files\SAS\SASOLAPCubeStudio\9.x**.

olapdef.txt

contains the user profile for SAS OLAP Cube Studio. The default storage location is **C:\Documents and Settings\user-id**.



APPENDIX

1

Recommended Reading

Recommended Reading 79

Recommended Reading

Here is the recommended reading list for this title:

- *SAS Intelligence Platform: Overview*
- *SAS Intelligence Platform: System Administration Guide*
- *SAS Intelligence Platform: Web Application Administration Guide*
- *SAS Intelligence Platform: Security Administration Guide*
- *SAS Intelligence Platform: Data Administration Guide*

For a complete list of administration documentation for the SAS Intelligence Platform, see <http://support.sas.com/913administration>.

For a list of SAS documentation, see

<http://support.sas.com/documentation/onlinedoc/sas9doc.html>.

For a complete list of SAS publications, see the current *SAS Publishing Catalog*. To order the most current publications or to receive a free copy of the catalog, contact a SAS representative at

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513
Telephone: (800) 727-3228*
Fax: (919) 677-8166
E-mail: sasbook@sas.com

Web address: support.sas.com/pubs

* For other SAS Institute business, call (919) 677-8000.

Customers outside the United States should contact their local SAS office.

Glossary

metadata profile

a client-side definition of where a metadata server is located. The definition includes a host name, a port number, and a list of one or more metadata repositories. In addition, the metadata profile can contain a user's login information and instructions for connecting to the metadata server automatically.

metadata repository

a collection of related metadata objects, such as the metadata for a set of tables and columns that are maintained by an application. A SAS Metadata Repository is an example.

metadata server

a server that provides metadata management services to one or more client applications. A SAS Metadata Server is an example.

OLAP (online analytical processing)

a software technology that enables users to dynamically analyze data that is stored in multidimensional database (MDDB) tables.

SAS Management Console

a Java application that provides a single user interface for performing SAS administrative tasks.

SAS Metadata Repository

a repository that is used by the SAS Metadata Server to store and retrieve metadata. See also SAS Metadata Server.

SAS Metadata Server

a multi-user server that enables users to read metadata from or write metadata to one or more SAS Metadata Repositories. The SAS Metadata Server uses the Integrated Object Model (IOM), which is provided with SAS Integration Technologies, to communicate with clients and with other servers.

SAS OLAP Cube Studio

a Java interface for defining and building OLAP cubes in SAS System 9 or later. Its main feature is the Cube Designer wizard, which guides you through the process of registering and creating cubes.

SAS OLAP Server

a SAS server that provides access to multidimensional data. The data is queried using the multidimensional expressions (MDX) language.

schema

a map or model of the overall data structure of a database. An OLAP schema specifies which group of cubes an OLAP server can access.

star schema

tables in a database in which a single fact table is connected to multiple dimension tables. This is visually represented in a star pattern. SAS OLAP cubes can be created from a star schema.

tuple

a data object that contains two or more components. In OLAP, a tuple is a slice of data from a cube. It is a selection of members (or cells) across dimensions in a cube. It can also be viewed as a cross-section of member data in a cube. For example, ([time].[all time].[2003], [geography].[all geography].[u.s.a.], [measures].[actualsum]) is a tuple that contains data from the Time, Geography, and Measures dimensions.

wizard

an interactive utility program that consists of a series of dialog boxes, windows, or pages. Users supply information in each dialog box, window, or page, and the wizard uses that information to perform a task.

Index

A

- access controls
 - for information maps 69
 - multiple inheritance of 28
- accessibility features 2
- Analytics Platform server 54
- Apache Tomcat HTTP server
 - customizing 61
- application server
 - defining for remote job processing 21
- ArcGIS software 52

B

- batch job deployment directory 23
- bulk loading data 36

C

- change management
 - metadata permissions 13
 - metadata profiles 14, 15
 - project repositories 12
 - repository directories 11
 - setting up 11
- client machine
 - setting up 19
- code transformations
 - importing and exporting 28
- configuration
 - IBM WebSphere MQ software 32
 - Microsoft MSMQ software 32
 - model management 58
 - SAS Data Quality Server 39
 - SAS Enterprise Miner 54
 - SAS Information Map Studio 70
 - SAS Management Console 75
 - SAS OLAP Cube Studio 78
- configuration files
 - SAS Enterprise Miner 56
- custom action 45
- Custom-tree folders
 - for security 25
 - multiple inheritance of access controls 28
 - setting permissions on 25
- Custom tree object types 27

D

- data quality options 42
- data servers
 - connecting to 10
- data sources
 - for information maps 69
- DBMS tables
 - bulk loading data into 36
- deployment
 - Enterprise client and Analytics Platform server 54
 - personal workstation 53

E

- e-mail action 45
- Enterprise client and Analytics Platform server
 - deployment 54
- Enterprise Miner
 - See* SAS Enterprise Miner
- Enterprise Miner plug-in for SAS Management Console 57
- ESRI maps
 - enabling in SAS Enterprise Guide 52
- exporting
 - generated transformations 28
 - metadata 29, 30

F

- folder level access 63
- FTP 46

G

- generated transformations
 - importing and exporting 28

H

- HTTP
 - retrieving files with 46

I

- IBM WebSphere MQ software 31
 - installing and configuring 32
- importing
 - generated transformations 28
 - information maps 69
 - metadata 29
- individual project level access 65
- information maps
 - access control 68, 69
 - data sources for 69
 - importing 69
 - storage management 68
- inheritance
 - multiple inheritance of access controls 28
- install directory
 - SAS Enterprise Miner 56
- installing desktop applications 2

J

- Java Access Bridge (JAB), installing 2
- Java heap
 - managing size 71
- Java Virtual Machine
 - properties 56
- Java Web Start properties 56
- job deployment
 - remote 24
- job processing
 - remote 20

L

- libraries
 - access for SAS Add-In for Microsoft Office 4
 - access for SAS Enterprise Guide 50
 - defining 18
- local host 17
 - setting up 17
- locales
 - downloading 41
- log files
 - managing in SAS Information Map Studio 71
- log information
 - routing to a file 43

M

- macro variables
 - SAS Enterprise Miner 55
- message queues 31
 - managing 33
 - registering metadata for 33
 - requirements for 31
- metadata
 - customizing for Enterprise Miner 57
 - defining tables for remote host 19
 - importing and exporting 29, 30
 - registering for message queues 33
 - securing for Enterprise Miner 63
 - setting permissions 13
 - supported formats 29
- metadata profiles 8, 15
 - creating 14
- metadata servers
 - connecting to 8
- Microsoft MSMQ software 31
 - installing and configuring 32
- Microsoft Office software 3
- model groups
 - creating 60
- model management configuration 58
- Models folder
 - SAS Enterprise Miner 60
- multi-tier environments
 - remote data access 17
 - SAS Data Integration Studio 16
- multiple inheritance
 - of access controls 28

N

- New Server wizard
 - retrieving files with FTP or HTTP 46

O

- object spawners
 - configuring for bulk loading 36
 - verifying 24
- OLAP cubes 77
 - enabling ESRI maps for 52
- output
 - redirecting 43

P

- permissions
 - setting metadata permissions 13
 - setting on Custom-tree folders 25
- personal workstation deployment 53
- Platform Computing Scheduling Server
 - testing 37
- project repositories
 - responding when none displayed 15
- project repository
 - creating 12
- Projects folder
 - SAS Enterprise Miner 58

- projects level access 64
- properties
 - Java Virtual Machine 56
 - Java Web Start 56

R

- remote data access 17
 - testing setup with View Data 20
- remote host 17
 - defining tables in metadata for 19
 - setting up 17
- remote job deployment 24
- remote job processing 20
- remote SAS/CONNECT server 18
- repository directory
 - creating 11
- routing log information 43

S

- SAS Add-In for Microsoft Office 3
 - connecting into SAS Intelligence Platform 3
 - library access 4
- SAS BI Web Services 34
 - installing 34
- SAS command
 - editing for workspace server 22
- SAS/CONNECT server
 - remote, setting up 18
- SAS Data Integration Studio 6
 - administrative tasks 7
 - change management 11
 - client machine 19
 - connecting to data servers 10
 - connecting to SAS servers 8
 - Custom-tree folders for security 25
 - data quality options 42
 - importing and exporting code transformations 28
 - importing and exporting metadata 29
 - logging information to a file 43
 - multi-tier environments 16
 - redirecting output 43
 - remote job deployment 24
 - remote job processing 20
 - responding when project repositories not displayed 15
 - setting default umask for 16
 - setting up SAS Data Quality Server 39
 - status code handling 45
 - testing Platform Computing Scheduling Server 37
- SAS Data Quality Server 39
 - configuring 39
 - creating schemes 42
 - data quality options 42
 - downloading locales 41
 - setting up 39
 - testing 40
- SAS Data Step Batch Server 23
- SAS Enterprise Guide 49
 - connecting into SAS Intelligence Platform 49
 - enabling ESRI maps 52

- library access 50
- SAS Enterprise Miner 53
 - configuration file 56
 - configuring 54
 - customizing Apache Tomcat HTTP server 61
 - customizing metadata 57
 - customizing servers for 59
 - deployment 53
 - Enterprise client and Analytics Platform server
 - deployment 54
 - folder level access 63
 - individual project level access 65
 - install directory 56
 - Java Virtual Machine properties 56
 - macro variables 55
 - Models folder 60
 - personal workstation deployment 53
 - Projects folder 58
 - projects folder level access 64
 - SAS Workspace Server level access 66
 - securing metadata 63
- SAS ETL Studio
 - See SAS Data Integration Studio
- SAS Information Map Studio 67
 - access controls for information maps 69
 - administrative tasks 67
 - configuration 70
 - data sources 69
 - importing information maps 69
 - input resources 68
 - logging on 70
 - managing size of the Java heap 71
 - managing storage for information maps 68
 - managing the log file 71
 - setting up users for 70
- SAS Intelligence Platform
 - connecting SAS Add-In for Microsoft Office into 3
 - connecting SAS Enterprise Guide into 49
 - setting up for SAS OLAP Cube Studio 78
- SAS Management Console 73
 - administrative tasks 73
 - configuration 75
 - Enterprise Miner plug-in for 57
 - Schedule Manager plug-in for 38
 - Server Manager plug-in for 46
 - starting for remote job processing 21
- SAS Object Spawner 22
- SAS OLAP Cube Studio 77
 - configuration 78
 - setting up SAS Intelligence Platform for 78
- SAS servers
 - connecting to 8
- SAS Workspace Server
 - See workspace server
- Schedule Manager 38
- scheduling
 - SAS Data Integration Studio jobs 37
- scheduling server
 - See Platform Computing Scheduling Server
- schemes
 - creating 42
- security
 - Custom-tree folders for 25
- server extensions 57
- servers
 - customizing for Enterprise Miner 59

status code handling 45
 custom action 45
 e-mail action 45

T

tables
 defining in metadata for remote host 19
testing
 Platform Computing Scheduling Server 37
 SAS Data Quality Server 40

U

umask
 setting default for SAS Data Integration Studio 16

users
 SAS Information Map Studio 70

V

View Data 20

W

Web services 34
 calling 36
 installing SAS BI Web Services 34

WebDAV server
 specifying with Models folder 60
WebSphere MQ software
 See IBM WebSphere MQ software
workspace server
 connecting to 9
 defining for remote job processing 21
 editing SAS command for 22
 enabling to execute operating system commands 37
 securing access for metadata at server level 66

Your Turn

If you have comments or suggestions about *SAS® 9.1.3 Intelligence Platform: Desktop Application Administration Guide*, please send them to us on a photocopy of this page, or send us electronic mail.

For comments about this book, please return the photocopy to

SAS Publishing
SAS Campus Drive
Cary, NC 27513
E-mail: yourturn@sas.com

For suggestions about the software, please return the photocopy to

SAS Institute Inc.
Technical Support Division
SAS Campus Drive
Cary, NC 27513
E-mail: suggest@sas.com

SAS® Publishing gives you the tools to flourish in any environment with SAS®!

Whether you are new to the workforce or an experienced professional, you need to distinguish yourself in this rapidly changing and competitive job market. SAS® Publishing provides you with a wide range of resources—including publications, online training, and software—to help you set yourself apart.

Expand Your Knowledge with Books from SAS® Publishing

SAS® Press offers user-friendly books for all skill levels, covering such topics as univariate and multivariate statistics, linear models, mixed models, fixed effects regression, and more. View our complete catalog and get free access to the latest reference documentation by visiting us online.

support.sas.com/pubs

SAS® Self-Paced e-Learning Puts Training at Your Fingertips

You are in complete control of your learning environment with SAS Self-Paced e-Learning! Gain immediate 24/7 access to SAS training directly from your desktop, using only a standard Web browser. If you do not have SAS installed, you can use SAS® Learning Edition for all Base SAS e-learning.

support.sas.com/selfpaced

Build Your SAS Skills with SAS® Learning Edition

SAS skills are in demand, and hands-on knowledge is vital. SAS users at all levels, from novice to advanced, will appreciate this inexpensive, intuitive, and easy-to-use personal learning version of SAS. With SAS Learning Edition, you have a unique opportunity to gain SAS software experience and propel your career in new and exciting directions.

support.sas.com/LE



**THE
POWER
TO KNOW®**

