SAS® 9.4 Intelligence Platform: System Administration Guide, Fourth Edition

Contents

PART 6   Managing SAS Metadata   225

PART 7   Using the Promotion Tools   263

PART 8   SAS Intelligence Platform Batch Tools: Reference

What's New in System Administration for the SAS 9.4 Intelligence Platform

Overview

The SAS Intelligence Platform includes the following enhancements and changes for SAS 9.4:

- a new administrative interface

- new servers

- SAS Metadata Server clustering

- changes to SAS server operation, monitoring, and logging

- new deployment backup and recovery facility

- new SAS Backup Manager user interface

- new batch utilities

- enhancements to promotion tools

- deprecated features

- documentation enhancements

New Administrative Interface: SAS Environment Manager

SAS Environment Manager is a new monitoring and management system for SAS deployments. Features include automatic resource discovery, monitoring of remote systems, personal and role-based dashboards, alerting, and visualization. The application provides web-based management, operation, and proactive monitoring of servers on both the middle tier and the SAS server tier. SAS Environment Manager incorporates some of the Hyperic technology from VMware in order to offer enterprise-class operational features.

Over the lifecycle of SAS 9.4, functions are being added to extend the capabilities of SAS Environment Manager as a centralized administration application for all SAS products. At this time, no functionality has been removed from the other SAS administrative applications such as SAS Management Console and the SAS Web Administration Console.

In SAS 9.4M1, SAS Environment Manager includes the ability to manage folders and to view and change authorization settings in SAS metadata. In SAS 9.4M2, SAS Environment Manager includes the ability to create and update access control templates (ACTs).

SAS Environment Manager 2.4 was shipped in October 2014. In this release, user administration features enable you to create and maintain users, groups, and roles. You can also manage memberships, logins, and internal accounts. This release also includes the SAS Environment Manager Service Management Architecture, which provides functions and capabilities that enable SAS Environment Manager to fit into a service-oriented architecture (SOA). These functions include SAS Environment Manager Extended Monitoring, Audit, Performance, and Measurement (APM) ETL; Agent-Collected Metric (ACM) ETL; and a solution kit framework.

SAS Environment Manager 2.5, which is provided with SAS 9.4M3, includes the following enhancements:

- Server and library metadata definitions can be viewed, and SAS LASR libraries, SAS LASR Servers, and Base SAS libraries can be created and edited.

- The stored process reports in the Report Center have been replaced with stored process prompts. In addition, a federated data mart enables you to collect metric data in data marts for several SAS deployments and view the collected metric data in one place.

- Log collection and discovery has been improved. Rather than relying on log locations that are stored in metadata, the ETL processes look through the directory structure of a SAS deployment to find log files.

- The new SAS Backup Manager is available on the **Administration** tab.

For details, see the *SAS Environment Manager: User's Guide*, *SAS Environment Manager 2.5 Administration: User's Guide*, *SAS Environment Manager 2.6 Administration: Fundamentals*, and the Help for SAS Environment Manager and SAS Environment Manager Administration.

New Servers

The middle-tier software includes a SAS Web Server for use as an HTTP server and a SAS Web Application Server, so that third-party web application servers are no longer needed or supported. SAS Web Application Server is a lightweight server that provides enterprise-class features for running SAS web applications. It is a specialized, extended configuration of Pivotal tc Server. The SAS deployment tools can configure these servers automatically. The tools simplify the configuration of vertical and horizontal clustering as well as HTTP load balancing.

Note: Prior to the February 15, 2022 release of SAS 9.4M7, SAS Web Application Server instances are managed with Pivotal tc Server. Thereafter, SAS Web Application Server instances are managed with Apache Tomcat.

The new SAS Web Infrastructure Platform Data Server replaces the SAS Framework Data Server as the default location for middle-tier data. This includes data such as alerts, comments, and workflows, as well as data for the SAS Content Server. The server is provided as an alternative to using a third-party DBMS. (The server cannot be used as a general-purpose data store.)

SAS Metadata Server Clustering

SAS 9.4 provides the option of implementing a metadata server cluster. This feature provides redundancy and high availability of the metadata server, which is a core component of the SAS infrastructure. Clustering ensures that the server will continue to operate if a server host machine fails.

A metadata server cluster is a group of three or more host machines (nodes) which have been configured as identical metadata servers. Each node runs its own server process and has its own server configuration information, journal file, and copy of the repository data sets. In addition, each node maintains a complete in-memory copy of the metadata repository. The nodes work together as if they were a single metadata server.

Client applications and users interact with the cluster in the same way that they would interact with a metadata server that is not clustered. A load-balancing process automatically distributes work among the nodes. If a node ceases to operate, the server continues to be available using the remaining nodes.

Effective with SAS 9.4M2, metadata server clustering is supported on z/OS. In previous SAS 9.4 releases, it is supported only on UNIX and Windows. All of the nodes in the cluster must have the same operating system.

SAS 9.4M3 includes a new metadata analysis and repair tool called Metadata Server Cluster Synchronization. This tool verifies that metadata is synchronized among all the nodes of a metadata server cluster. For details, see "Analyzing and Repairing Metadata" on page 236.

For more information, see Chapter 16, "Using Metadata Server Clustering," on page 211.

Changes to SAS Server Operation, Monitoring, and Logging

SAS 9.4 includes the following changes to operation, monitoring, and logging for SAS servers:

- The sas.servers UNIX script includes all SAS servers that are needed at start-up, including middle-tier as well as server-tier processes. See "Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers" on page 67.

- In SAS 9.4M1, the documentation was changed to recommend that the SAS Web Server be started before the SAS Web Application Server. This start-up order helps ensure optimum performance when web applications are initialized. The sas.servers script has also been changed to incorporate the new order.

- In SAS 9.4M3, the documentation was changed to recommend starting the SAS Metadata Server first, followed by the SAS Web Infrastructure Platform Data Server (instead of the reverse). This order is easier and more logical for deployments in which the metadata server is installed on a separate machine from the other server-tier components.

- In SAS 9.4M1, the metadata server script includes a –recover option. This option starts a server that is not currently running, and then restores the server's metadata

repository from the most recent backup. The option provides an easy way to recover a server or node that is unresponsive. See "Additional Script Options for the SAS Metadata Server" on page 74.

- The server management features of SAS Management Console have been enhanced as follows:

  - The object spawner can be refreshed for a specific server context, and SAS/CONNECT spawners can now be managed using the server management features.

  - Performance counters can be reset for the metadata server, an OLAP server, or an object spawner, and for pooled workspace server and stored process server sessions.

  - A new option enables you to clear the decision cache for the object spawner and SAS/CONNECT spawner.

  See "Using SAS Management Console to Operate SAS Servers" on page 76 and "Using SAS Management Console to Monitor SAS Servers" on page 112.

- The Metadata Manager in SAS Management Console includes the following new features:

  - The metadata server can be paused to Read Only mode. See "Use SAS Management Console to Operate the SAS Metadata Server" on page 76.

  - The Active Server Properties dialog box contains a new **Options** tab that displays metadata server configuration properties. For clustered servers, the dialog box also contains a **Cluster** tab that displays the cluster configuration and status. See "Use SAS Management Console to Check the Access State and Other Properties of the SAS Metadata Server" on page 88.

- The following options have been added to the metadata server batch command:

  - an option that starts the server in offline mode

  - an option that stops all of the nodes in a clustered metadata server

  - in a clustered metadata server, an option that starts a single node without clustering

  See "Use Individual Scripts to Operate Servers" on page 70.

- The following changes have been made to SAS Intelligence Platform logging:

  - The Perf.Meta.Expensive logger logs requests that take longer than a specified time threshold so that application developers and administrators can identify high-cost metadata requests.

  - A new logger has been added for resolving issues with library pre-assignment.

  - New loggers have been added to audit changes that are made to a server's logging configuration file.

  - Potentially sensitive loggers for workspace servers, pooled workspace servers, and stored process servers are set to immutable by default.

See Chapter 10, "Administering Logging for SAS Servers," on page 131.

New Deployment Backup and Recovery Tool

The new Deployment Backup and Recovery tool provides an integrated method for backing up and recovering your SAS content across multiple tiers and machines. The following components on the SAS server tier, middle tier, and data tier are included in the backup:

- the metadata server

- the Data directories, SASEnvironment directories, and server configuration directories for each server on the SAS server tier

- the SAS Content Server repository

- databases that are managed by the SAS Web Infrastructure Platform Data Server

- additional directories that you specify

By default, the Deployment Backup and Recovery tool backs up these items automatically each Sunday at 1:00 a.m. If you specify a central vault location, the backups from each host machine are copied to that location following each backup operation.

A set of batch commands is provided that enable you to do the following:

- modify the backup schedule, the backup retention period, and the vault location

- specify additional directories to include in the backup

- execute ad hoc backups

- perform a recovery from a backup

- display backup status, history, schedule, and configuration information

Effective with SAS 9.4M1 the Deployment Backup and Recovery tool includes the following changes:

- The syntax has changed for the sas-backup and sas-list-backups commands.

- The input file specifications have changed for the following commands: sas-backup, sas-set-backup-schedule, sas-remove-backup-schedule, sas-update-backup-config, sas-add-backup-customdir, and sas-remove-backup-customdir.

- The sas-status-backup command can be used only to display the status of a backup operation. It no longer reports the status of recoveries.

- An alert email is generated if a backup or recovery is unsuccessful. By default, the email is sent to the system administrator email address that was specified in the SAS Deployment Wizard.

- A new offline recovery command (sas-recover-offline) can restore files from the deployment backup when some resources are unavailable or have been taken offline.

- On UNIX machines, it is no longer necessary to list the backup user (generally the SAS Installer user) in the etc/sudoers file.

Effective with SAS 9.4M2, the Deployment Backup and Recovery tool includes the following changes:

- If you have specified a central vault location, the local backups are deleted from each host machine after they are successfully copied to the central vault. However,

metadata server backups are not deleted from the location that is used by the metadata server backup utility.

- A central vault location is no longer required if your middle-tier environment includes a clustered SAS Content Server.

- When you use sas-update-backup-config to specify a central vault location, the location is automatically added to the BackupServer.ini file.

- The syntax has changed for sas-remove-backup-schedule and sas-remove-backup-customdir. JSON files are no longer used with these commands.

- The input data specifications for sas-add-backup-customdir have changed.

- Error messages have been modified to improve clarity.

- The recovery process uses checksums to validate the contents of the specified backup directory against the originally created backup. If the validation fails, a warning message is generated and the recovery is not performed. The **forced** option can be specified in the recovery command's JSON file to perform the recovery without validating the files.

- Commands can be run from any directory.

Effective with SAS 9.4M3, the Deployment Backup and Recovery tool includes the following changes:

- The following commands have been deprecated: sas-display-backup-source-config, sas-add-backup-customdir, sas-display-backup-customdir, and sas-remove-backup-customdir.

- The sas-update-backup-config command includes these new capabilities:

  - You can include or exclude specific tiers, specific instances of the SAS Web Infrastructure Platform Data Server, or particular databases that are managed by that server.

  - You can specify that the metadata server's metadata repositories are to be reorganized during the backup.

  - You can define filters that specify which subdirectories and files are to be included or excluded when backing up the **Data**, **SASEnvironment**, and server configuration directories for a particular **Levn** instance.

  - You can specify additional (custom) directories to be backed up under a particular **Levn** instance. In addition, you can define filters that specify which subdirectories and files are to be included or excluded from these backups

- The command sas-display-backup-config has a new -sourcetype option and a new output format. In addition, it creates a JSON file that describes the current backup configuration, which is useful when you want to define a filter.

- The command sas-status-backup displays the message **purged: True** (instead of **offlineBackup: True**) if backup files have been purged due to the retention policy.

- It is no longer necessary to perform manual steps to complete the recovery of the SAS Content Server.

- Some of the labels in the input file for sas-recover-offline have been changed.

- The input file for sas-recover-offline includes an option to include the metadata server's configuration files in the recovery.

For more information, see Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

New SAS Backup Manager User Interface

SAS 9.4M3 includes SAS Backup Manager, an easy-to-use interface for scheduling, configuring, monitoring, and performing integrated backups. The interface incorporates most of the functions of the Deployment Backup and Recovery tool's batch commands. SAS Backup Manager can be accessed from the **Administration** tab of SAS Environment Manager. See "About SAS Backup Manager" on page 391.

New Batch Utilities

In addition to the Deployment Backup and Recovery batch tools, the following batch tools have been added to the Platform Object Framework:

- Metadata management tools, which you can use for tasks such as listing selected objects, deleting selected objects, creating new folders, and managing metadata access. See Chapter 24, "Batch Tools for Metadata Management," on page 315.

- Administrator tools for tasks such as backing up the metadata server, creating and deleting metadata repositories, and updating metadata profiles. See Chapter 27, "Using the Batch Tools for Metadata Server Administration," on page 375.

- Batch tools for relationship reporting. These tools, which are new with SAS 9.4M1, enable you to identify relationships among the content objects in the SAS Folder tree. For example, you can identify the objects that a given object depends on or contains; the objects that depend on or contain a given object; and the objects that are associated with a given object. Both direct and nested relationships are identified. See Chapter 26, "Using the Batch Relationship Reporting Tools," on page 347.

  SAS 9.4M2 includes the following changes to the relationship reporting tools:

  - The terminology on the reports is slightly different. For example, the term **Contains** replaces the term **Includes**.

  - Keywords and responsible user have been added to the available filtering arguments.

  SAS 9.4M3 includes the following changes to the relationship reporting tools:

  - Automatic loading of relationship data is configured by default to execute on an hourly basis. Cleaning of relationship data is configured by default to execute daily at 11:00 p.m

  - On impact reports, the phrase **Is dependent on** has been replaced with the word **Impact**.

  - A new batch command enables you to use SAS Metadata Bridges to load third-party lineage information to the SAS Relationship Service. See "Using the SAS Metadata Bridge Relationship Loader (sas-metabridge-relationship-loader)" on page 368.

- Analyze and Repair Metadata batch tool. This tool, which is new with SAS 9.4M2, provides a batch method for running the analyze and repair tools that are available in the Metadata Manager node in SAS Management Console. You use the tool to run selected tests on metadata to locate common problems and, when possible, to repair

problems that the analysis has identified. See "Analyze and Repair Metadata (sas-analyze-metadata)" on page 382.

Enhancements to Promotion Tools

The following enhancements have been added to the promotion tools. These tools include the Export SAS Package and Import SAS Package wizards in SAS Management Console, SAS Data Integration Studio, and SAS OLAP Cube Studio, as well as the batch import and export tools.

- Machine names in stored user IDs are updated to the target metadata server machine name during the import process.

- When you promote an HTTP server, the base path can be edited.

- Effective with SAS 9.4M2, a new global option is available for the batch import tool. When you import table metadata, this option enables you to exclude metadata for any columns that already exist in the target table.

- Effective with SAS 9.4M3, you can use the -disableX11 option to run the batch import or export tool on UNIX without setting the DISPLAY variable.

See "Promotion Details for Specific Object Types" on page 292.

Viewing Setinit Information In Metadata

The metadata repository stores setinit (license) information for some SAS solutions that depend on the SAS middle tier. Effective with SAS 9.4M2, you can use SAS Management Console to view setinit information that is stored in metadata. See Chapter 19, "Managing Setinit (License) Information in Metadata," on page 261.

Deprecated Features

The following features have been removed:

- The SASMeta server context no longer includes a workspace server and a batch server. These servers are no longer needed by metadata utilities.

- The SAS Framework Data Server has been replaced by the SAS Web Infrastructure Platform Data Server.

- The SAS Remote Services application is deprecated and SAS Enterprise Business Intelligence deployments do not use the server. It is included with the software for backward compatibility with custom applications that rely on it.

- SAS has deprecated support of the SAS Audit, Performance and Measurement package. The functionality provided by the package was integrated into SAS Environment Manager 2.4.

Documentation Enhancements

A new section titled "SAS Intelligence Platform Batch Tools: Reference" contains documentation for the new batch utilities as well as for the batch export and import tools. In the third edition (published in April 2014), documentation for the batch relationship reporting tools was added to this section.

Accessibility

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

Part 1

Getting Started

Chapter 1

Introduction to the SAS Intelligence Platform: System Administration Guide

About the SAS Intelligence Platform: System Administration Guide

The SAS Intelligence Platform is a comprehensive, end-to-end infrastructure for creating, managing, and distributing enterprise intelligence. The SAS Intelligence Platform comprises several software offerings, including the following:

• SAS BI Server

• SAS Enterprise BI Server

• SAS Enterprise Data Integration Server (for renewals only) and SAS Data Integration Server

• SAS Data Management (Standard or Advanced)

Building on the technologies in the SAS Intelligence Platform, SAS offers solutions for industries such as financial services, life sciences, health care, retail, and manufacturing, as well as line-of-business solutions in areas such as customer management, enterprise risk management, and financial management. Therefore, if your SAS deployment plan includes a SAS solution, then the SAS Intelligence Platform documentation also pertains to you.

This guide covers administrative tasks that you will need to perform on an ongoing basis after the initial installation and configuration of the SAS Intelligence Platform. The guide covers tasks that apply to the SAS Intelligence Platform as a whole, including the following:

• starting and stopping servers

• checking the status of servers

• monitoring server activity

• administering server logging

• administering the SAS Metadata Server, including the following tasks:

  • performing backups and restores

- monitoring the server

- optimizing the server's performance

- configuring and managing clustered servers

- administering SAS Metadata Repositories and metadata content, including the following tasks:

  - creating repositories

  - analyzing and repairing metadata

  - working with SAS Folders

  - promoting metadata and related physical content

- using the batch tools that are provided for various metadata management and administration tasks

The appendixes of this guide provide reference information about log files, configuration files, and default ports. They also provide information about utilities that you can use to modify host name references and to migrate content to another SAS 9.4 deployment.

This guide is one of several documents that SAS publishes to support administration of the SAS Intelligence Platform. Additional guides are available to cover the following administration topics:

- SAS Application Server administration

- data administration

- security administration

- middle-tier administration

- web application administration

- desktop application administration

- scheduling

- logging

To access these documents, go to **http://support.sas.com/ 94administration**.

Additional information for specific operating systems is available in the SAS companion guides.

Prerequisites for Using the SAS Intelligence Platform: System Administration Guide

Before using this guide, you should be familiar with the concepts and terminology that are introduced in the *SAS Intelligence Platform: Overview*.

In addition, you should have completed all of the tasks that are described in the *SAS Intelligence Platform: Installation and Configuration Guide*.

To access these documents, go to **http://support.sas.com/ 94administration**.

Chapter 2

Understanding the State of Your System

Overview of the State of Your System

Before you begin administering the SAS Intelligence Platform, you should be aware of the following components that have been put in place during the installation and deployment process:

- the SAS Metadata Server, which serves metadata from one or more SAS Metadata Repositories to all SAS Intelligence Platform client applications.

- the SAS Application Server components (also referred to as compute servers) that have been installed and configured in your environment to run SAS processes. These components can include SAS Workspace Servers, SAS Pooled Workspace Servers, SAS Stored Process Servers, SAS OLAP Servers, SAS/CONNECT servers, SAS Grid Servers, and SAS batch servers.

  Note: In the SAS Intelligence Platform, the term server refers to a program or programs that wait for and fulfill requests from client programs for data or services. The term server does not necessarily refer to a physical computer, since a single computer can host one or more servers of various types.

- other SAS servers that might have been installed in your environment. These servers might include a SAS Web Application Server, a SAS Web Server, a SAS Web Infrastructure Platform Data Server, a SAS Content Server, and a SAS/SHARE server.

- the ports that are used by each server to listen for incoming requests.

- the configuration directories that store configuration files, logs, scripts, and special-purpose SAS data sets on each SAS server machine and each middle-tier machine.

- the main configuration files that control the operation of the system.

- the initial SAS users, groups, and roles that have been defined, both on your host operating systems and in the SAS Metadata Repository.

Overview of the SAS Metadata Server and Its Initial Configuration

What Is the SAS Metadata Server?

The SAS Metadata Server is the most critical software component in the SAS Intelligence Platform. All of the client applications and the other SAS servers that are part of the platform depend on the SAS Metadata Server and cannot operate without it.

The SAS Metadata Server is a multi-user server that serves metadata from one or more SAS Metadata Repositories to all of the SAS Intelligence Platform client applications in your environment. The SAS Metadata Server enables centralized control so that all users access consistent and accurate data.

The functionality of the SAS Metadata Server is provided through the SAS Open Metadata Architecture, which is a metadata management facility that provides common metadata services to applications. One metadata server supports all of the SAS

applications in your environment and can support hundreds of concurrent users. This architecture enables the following:

- the exchange of metadata between applications, so that applications can work together more easily.

- centralized management of metadata resources. Because there is a common framework for creating, accessing, and updating metadata, it is easier to manage the applications that rely on this metadata.

The SAS Metadata Server stores information about the following:

- the enterprise data sources and data structures that are accessed by SAS applications

- resources that are created and used by SAS applications, including information maps, OLAP cubes, report definitions, stored process definitions, and scheduled jobs

- the servers that run SAS processes

- the users and groups of users that use the system, and the levels of access that users and groups have to resources

Effective with SAS 9.4, the SAS Metadata Server can be set up in a clustered configuration to provide redundancy and high availability.

About the Metadata Manager Plug-in

In SAS Management Console, the SAS Metadata Server appears in the Metadata Manager plug-in, as shown here:
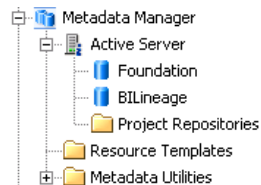
You can use this plug-in to stop, pause, resume, or reset the metadata server; to check the status of the metadata server and the metadata repositories; and to create and manage metadata repositories. You can also access utilities such as the metadata server backup and recovery facility.

Note: In the Plug-ins view, the **Properties** action generally displays an object's metadata definition. However, when you select **Properties** on the **Active Server** node of the Metadata Manager plug-in, the information that is displayed is supplied directly by the metadata server.

About the SASMeta Server Context

So that you can make use of SAS Management Console's server monitoring features, the metadata server also appears in the SASMeta application server tree (also referred to as the SASMeta server context) in the Server Manager plug-in. The following example shows a metadata server that has been set up in a clustered configuration consisting of three nodes.
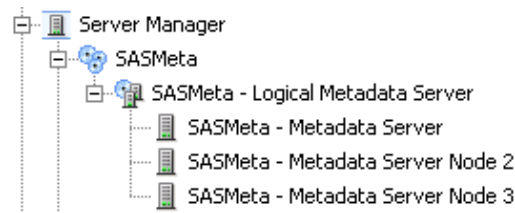
Metadata server nodes also appear in the **Resources** tab of SAS Environment Manager, which is new with SAS 9.4.

About the SAS Metadata Repositories and the Repository Manager

The SAS Metadata Server is a SAS process that enables clients to read metadata from and write metadata to one or more SAS Metadata Repositories. A repository is a physical location in which a collection of related metadata objects is stored.

Metadata repositories are managed by a repository manager, which is a library of tables that define the set of repositories that can be accessed by a SAS Metadata Server. A metadata server cannot be started without a repository manager. Each metadata server can have only one repository manager.

The SAS Metadata Server supports three types of metadata repositories:

foundation repository
> the required metadata store for a SAS Metadata Server. Each metadata server has one foundation repository. As a default, the name of the foundation repository is Foundation. (If you performed a custom installation, then you might have assigned a different name.)

project repository
> a metadata repository that serves as an isolated work area or playpen. Project repositories are available for SAS Data Integration Studio only. In general, each user who participates in change management has his or her own project repository.
>
> A project repository enables a user to check out and lock metadata from a foundation repository or a custom repository so that the metadata can be modified and tested in a separate area. When the metadata is ready, it is checked in and unlocked so that it is available to other users.

custom repository
> an optional metadata store for a SAS Metadata Server, in addition to the foundation repository. Custom repositories are useful for physically segregating metadata for storage or security purposes.
>
> Custom repositories appear as separate root folders in the **Folders** tab of SAS Management Console.

In SAS Management Console, the repositories are displayed in the Metadata Manager plug-in, as shown in this example:
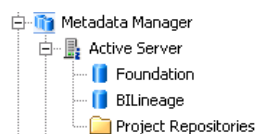
How the SAS Metadata Server Works

SAS Intelligence Platform applications connect to the SAS Metadata Server and issue SAS Open Metadata Interface method calls that access SAS Metadata Repositories.

To enable high-speed access by clients, the SAS Metadata Server is an "in-memory" server. As clients submit queries and updates, the data sets that contain the requested records are read from repository data sets on disk into the server's memory, where they remain until the server is paused and resumed or until the server is stopped.

In the default configuration, metadata updates that are requested by client applications are recorded immediately in memory and in a journal file so that client access can be resumed. The more time-consuming updates to the repository data sets on disk are performed later in the background.

About Connection Profiles

During initialization, SAS desktop applications use a connection profile to connect to the metadata server. A successful connection enables secure access to the enterprise resources that are defined in metadata. On the hosts of SAS desktop applications, connection profiles are stored in files with the extension **.swa** and are located in the following paths:

- Windows Vista or later operating environments: **C:\Users*user-name*\AppData\Roaming\SAS\MetadataServerProfiles**

- Other Windows systems: **C:\Documents and Settings*user-name*\Application Data\SAS\MetadataServerProfiles**

- UNIX systems: **/*user's-home-direcory*/.SASAppData/MetadataServerProfiles**

At installation time, most desktop applications create default connection profiles. Users can change, edit, or delete connection profiles at any time.

The Initial Configuration of the SAS Metadata Server

The SAS Deployment Wizard provides the following initial configuration for your metadata server:

- Your metadata server contains a foundation metadata repository called Foundation. (If you performed a custom installation, then you might have assigned a different name.) This repository contains the following items:

    - metadata definitions for initial users, groups, and roles

    - default authorization settings that give the initial users, groups, and roles access to different levels of functionality

    - metadata definitions for your servers

    - a full set of resource templates that can be used to create additional metadata objects

  The data sets that make up the foundation repository are typically located in the following path:

  SAS-configuration-directory/Lev1/SASMeta/MetadataServer/MetadataRepositories/Foundation

In this path, *SAS-configuration-directory* refers to the SAS configuration directory. See "About the Configuration Directory" on page 14.

- By default, metadata server backups are scheduled to run at 1:00 a.m. server local time every day except Sunday. The Deployment Backup and Recovery tool, if it is configured, backs up the metadata server (along with other resources) each Sunday at 1:00 a.m. by default. For details, see "Overview of the SAS Metadata Server Backup and Recovery Facility" on page 169.

- Roll-forward journaling has been enabled. This option creates a linear journal file that permanently stores all transactions that have occurred since the most recent backup. These transactions can be used to perform roll-forward recovery of the metadata server to a specified point in time after the last backup was taken.

- Two special users of the metadata server have been set up: the SAS Administrator and the SAS Trusted User.

- Metadata server events are written to a log file that is rolled over daily. The default location for logs is as follows:

  SAS-configuration-directory/Lev1/SASMeta/MetadataServer/Logs

  For details about the default logging configuration, see "Initial Logging Configuration for SAS Servers" on page 135.

Overview of SAS Application Server Components and Their Initial Configuration

About the SAS Application Server Hierarchy

The SAS Deployment Wizard creates metadata definitions and configuration files for a set of SAS servers that are referred to collectively as a SAS Application Server. The server hierarchy includes the following levels:

SAS Application Server
> a logical framework (or server context) under which SAS applications execute. This framework enables you to specify metadata that applies to all of the logical servers and servers that the SAS Application Server contains. You can also attach libraries, schemas, directories, and other resources that are available to SAS servers, regardless of the type of server. As a result, you can deploy applications in several modes while ensuring that applications will execute properly in each mode.
>
> The default name of the initial SAS Application Server is **SASApp**. You can specify a different name when you run the SAS Deployment Wizard.

logical servers
> a logical framework within SAS Application Server metadata that contains definitions for one or more similar servers. A logical server is referenced when an application requests a specific launch mode, such as interactive or batch.

servers
> specific process instances that perform the requested work. A server definition contains the server metadata that is required to connect to a SAS server on a particular machine. The definition includes the server's launch command, details about where the process is executing and how a client should contact the server, and options that describe how the server should behave.

Note: In the SAS Intelligence Platform, the term server refers to a program or programs that wait for and fulfill requests from client programs for data or services. The term server does not necessarily refer to a physical computer, since a single computer can host one or more servers of various types.

The SAS Application Server hierarchy appears in SAS Management Console in the Server Manager plug-in, as in the following example:
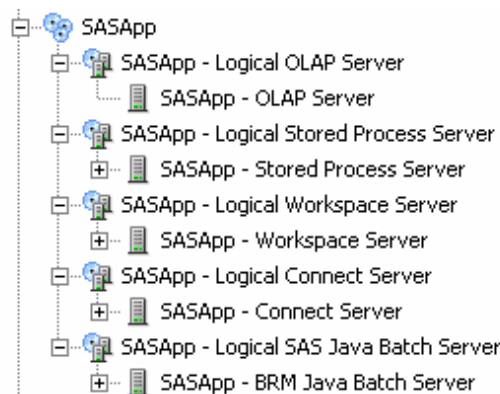
The SAS Application Server components also appear in the **Resources** tab of SAS Environment Manager.

About the SAS Application Server Components

The most common components of a SAS Application Server are SAS Workspace Servers, SAS Pooled Workspace Servers, and SAS Stored Process Servers. An application server can also contain SAS/CONNECT servers, SAS OLAP Servers, SAS DATA step batch servers, and SAS Grid Servers. The purpose of each server component is described in the following table.

Table 2.1 SAS Application Server Components

Server Name	Server Description
SAS Workspace Server	SAS Workspace Servers fulfill client requests for SAS sessions. When these servers are configured as standard workspace servers (without workspace pooling), each client establishes a connection to a single-user server process, uses the server, and then disconnects. For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.
SAS Pooled Workspace Server	SAS Pooled Workspace Servers are workspace servers that use server-side pooling. The SAS object spawner maintains a pool of reusable workspace server processes called a pool, thus avoiding the overhead that is associated with creating a new process for each connection. Workspace server load balancing is built in to server-side pooling.
	Pooled workspace servers are intended for use by query and reporting tools such as SAS Web Report Studio and SAS Information Delivery Portal when accessing relational information maps.
	For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.

Server Name	Server Description
SAS Stored Process Servers	SAS Stored Process Servers fulfill client requests to execute SAS stored processes. A stored process is a SAS program that is stored on a server and can be executed as required by requesting applications.
	Stored process servers are configured to be load balanced. By default, the object spawner balances workloads across three stored process server processes.
	For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.
SAS/CONNECT server	SAS/CONNECT servers enable clients to execute code on a remote host, or to move data between client and server machines. For details, see *SAS/CONNECT User's Guide*.
SAS OLAP Server	SAS OLAP Servers provide access to multidimensional data. For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.
SAS batch server	SAS batch servers are part of the configuration for SAS scheduling. They are metadata objects that store information about an executable that the scheduling server can use to run scheduled tasks. For details, see *Scheduling in SAS*.
SAS Grid Server	A SAS Grid Server is configured if you have the SAS Grid Manager software application. SAS Grid Servers are metadata objects that store information about an executable that the scheduling server can use to start SAS/CONNECT on grid nodes. For details, see *Grid Computing in SAS*.

Overview of Other SAS Servers and Spawners

The SAS Intelligence Platform includes the following additional servers and spawners that might appear in the Server Manager tree in SAS Management Console:

Table 2.2 Other SAS Servers and Spawners

Server or Spawner Name	Server or Spawner Description
SAS Content Server	Stores digital content (such as documents, reports, and images) that are created and used by SAS client applications. To interact with the SAS Content Server, clients use WebDAV-based protocols for access, versioning, collaboration, security, and searching.
	The SAS Content Server is used by SAS Information Delivery Portal and SAS Web Report Studio, as well as other SAS products and solutions.
	The server's repositories are stored in a database. For more information, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Server or Spawner Name	Server or Spawner Description
DataFlux Data Management Server	Executes DataFlux jobs and real-time services that cleanse data or analyze data quality. The jobs and services are created using DataFlux applications (dfPower Architect and dfPower Profile) and are executed on the server through job flows that are created in SAS Data Integration Studio.
	For information about configuring this server, see "Administering SAS Data Integration Studio" in *SAS Intelligence Platform: Desktop Application Administration Guide*.
SAS In-Process Services	Provides the ability to schedule jobs from certain web-based SAS applications without using a separate scheduling server. The scheduling functions run as a process within the application. For details, see *Scheduling in SAS*.
SAS object spawner	Runs on the same machine as the SAS Application Server components. The spawner listens for incoming client requests for workspace servers, pooled workspace servers, and stored process servers, and it launches instances of these servers as needed.
	Although the object spawner is not part of the SAS Application Server hierarchy, it is essential to the operation of workspace servers, stored process servers, and pooled workspace servers.
	For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.
SAS Web Infrastructure Platform Data Server	Is the default location for middle-tier data such as alerts, comments, and workflows, as well as data for the SAS Content Server. The server is provided as an alternative to using a third-party DBMS. (The server cannot be used as a general-purpose data store.)
	This server replaces the SAS Framework Data Server, which was included in SAS 9.3.
SAS/CONNECT spawner	Listens for SAS/CONNECT client requests and invokes SAS sessions on the remote computer. For details, see *SAS/CONNECT User's Guide*.
SAS/SHARE server	SAS/SHARE enables multiple users to update a SAS file (or several files) at the same time. It also enables users to access SAS files on a server without requiring a separate SAS/CONNECT report login for each user. For details, see the *SAS/SHARE User's Guide*.

In addition, the following server appears under the Deployment Tester plug-in under Application Management:

SAS Deployment Tester Server
  processes validation tasks that are requested either by the SAS Deployment Tester plug-in or by a SAS Deployment Tester batch client. For usage instructions, see Chapter 8, "Using the Deployment Tester," on page 97 and the SAS Management Console Help.

If your deployment includes a middle tier, then a SAS Web Application Server and SAS Web Server are also provided. For details, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Overview of Ports

The servers in the SAS Intelligence Platform communicate with clients and other servers using TCP/IP. Thus, each server uses a particular port or ports to listen for incoming requests or to communicate outgoing requests. For information about the default ports and their data direction, see the *SAS Intelligence Platform: Installation and Configuration Guide* and your pre-installation checklist. During installation, the SAS Deployment Wizard enables you to override the default port numbers with different values. Be sure to record these values on your pre-installation checklist.

Overview of the Configuration Directory Structure

About the Configuration Directory

Every server-tier and middle-tier host has a configuration directory which is the root of the SAS Intelligence Platform environment. The structure and contents of the directory vary depending on the host operating system, which products are installed, and whether the host machine is a server-tier host or a middle-tier host.

Throughout the SAS Intelligence Platform administration documentation, the variable *SAS-configuration-directory* is used to designate your configuration directory path. The default paths are as follows:

- on Windows systems:

  *drive:***\SAS***deployment-plan-name*

- on UNIX systems:

  /usr/local/*deployment-plan-name* or **/usr/local/sas**

  Note:

  - In these paths, *deployment-plan-name* is the name that was specified when the deployment plan was created. If no name was specified, then the default name **Config** is used.

  - You can specify a different path when you run the SAS Deployment Wizard. In UNIX environments, the installer generally overrides the path with the site's preferred location (for example, **/opt/sas/Config**). The installer must have Write permission to this path. Do not specify a path that is within the SAS installation directory (for example, **SASHome**).

- on z/OS systems, the directory that is specified in the **CONFIG_DIR** environment variable

Here is an example of the configuration directory structure for a single-machine Windows implementation:

```
☐ 📁 SAS
    ☐ 📁 Config
        ☐ 📁 Lev1
            ⊞ 📁 AppData
            ⊞ 📁 Applications
                📁 BILBatch
                📁 ConfigData
            ⊞ 📁 DeploymentTesterServer
            ⊞ 📁 Documents
            ⊞ 📁 Logs
            ⊞ 📁 ObjectSpawner
                📁 ReportBatch
            ⊞ 📁 SASApp
            ⊞ 📁 SASMeta
            ⊞ 📁 SchedulingServer
            ⊞ 📁 Temp
                📁 Utilities
            ⊞ 📁 Web
            ⊞ 📁 WebInfrastructurePlatformDataServer
                📁 WIPSchedBatch
            📁 Utilities
```
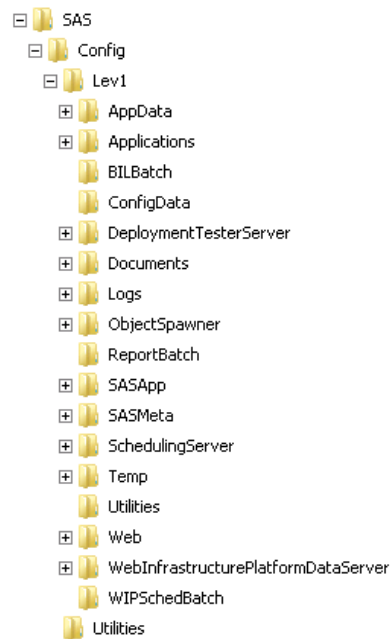
Note: Depending on your operating system and which products you have installed, your site might have different subdirectories.

About the Levn Subdirectory

The **Levn** subdirectory contains configuration information and other files for a particular installation instance. **Lev1** is generally used for production environments. Additional levels, such as **Lev2** and **Lev3**, can be used for environments that you install for purposes such as development and testing. During installation, the SAS Deployment Wizard enables you to select the level number.

The contents of the **Levn** directory depend on which components are installed on the machine. Its contents might include the following:

- on UNIX and z/OS machines on the server tier and the middle tier, a script called sas.servers that you can use to start, stop, or restart all of the SAS servers and spawners on in the correct order by using a single command.

- an **AppData** subdirectory, which contains indexes and the repository configuration file for the SAS Content Server. It also contains data that is installed for the use of specific applications (for example, SAS BI Dashboard).

- a subdirectory for each application server.

- a subdirectory for the metadata server.

- subdirectories for other SAS servers and spawners, including the SAS object spawner, the SAS Deployment Tester Server, the SAS Web Infrastructure Platform Data Server, the SAS/CONNECT spawner, and the SAS/SHARE server.

- a **Web** subdirectory for files related to web applications.

- logs and documents that were created during the installation and configuration process.

For more details about the **Levn** directory, see "Contents of the Levn Subdirectory" on page 440.

About the Application Server Subdirectory

On machines where components of a SAS Application Server are installed, the **Levn** directory contains an application server subdirectory. A SAS Application Server is a logical entity that contains specific servers (for example, a SAS Workspace Server and a SAS Stored Process Server) that execute SAS code. Application server subdirectories are named as follows:

- **SASApp** is the default name that the SAS Deployment Wizard assigns to directory for the first application server. If you assigned a different name when you ran the SAS Deployment Wizard, then the subdirectory is named accordingly.

- If you configure more than one application server, then the SAS Deployment Wizard creates additional directories that correspond to each of the application server names.

The application server subdirectory contains the following items:

- a subdirectory for each component of the application server, including the workspace server, the pooled workspace server, the OLAP server, the stored process server, the SAS/CONNECT server, batch servers, and grid servers (if applicable). Each of these subdirectories contains the server's start-up scripts, configuration files, logconfig.xml files to configure logging, and **Logs** directories to store log files.

- a subdirectory for data that is used by the application server.

- a **SASEnvironment** subdirectory that you can use to store deployed jobs, job steps, SAS format and informat catalogs, and user-created SAS autocall macros that are needed to perform processing on the application server

- configuration files that apply to all components of the application server

For more details about this subdirectory, see "Contents of the Application Server Subdirectory (SASApp)" on page 443. For information about the server configuration files that are contained in this subdirectory, see "Configuration Files for Components of SAS Application Servers" on page 450.

About the SASMeta Subdirectory

On the machines where the metadata server is installed, the **Levn** directory includes a subdirectory for the metadata server's files. **SASMeta** is the name that the SAS Deployment Wizard assigns to the metadata server context and, in turn, to this directory.

The **SASMeta** directory contains a **MetadataServer** subdirectory, which contains the following items:

- the metadata server's start-up and management scripts, configuration files, logconfig.xml files to configure logging, and **Logs** directories to store log files.

- a script to back up the metadata server (for details, see "SAS Metadata Server Backup Tasks" on page 173).

- a subdirectory called **MetadataRepositories**, which contains a subdirectory for each metadata repository. These subdirectories contain the SAS data sets that store your metadata.

- a subdirectory called **rposmgr**, which contains data sets for the repository manager.

  CAUTION:
  The MetadataRepositories and rposmgr directories contain data sets that are critical to the operation of your system.

To safeguard the integrity of your system:

- Never move, delete, modify, or directly read the data sets in these directories. These data sets should be accessed only by metadata server processes.

- Do not place other files in these directories.

- Back up the contents of these directories regularly, as described in Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153.

- a **Backups** directory that is the default destination for metadata server backups and their associated journal files.

For more information about the metadata server configuration files, see "Configuration Files for the Metadata Server" on page 448. For a complete list of the contents of the MetadataServer subdirectory, see "Contents of the Metadata Server Subdirectory (SASMeta/MetadataServer)" on page 441.

About the Web Subdirectory

On middle-tier machines (that is, machines with configured web components), the Lev1 directory contains a subdirectory called **Web**. For details, see "Contents of the Web Subdirectory" on page 444. For information about administering middle-tier applications and components, see the *SAS Intelligence Platform: Web Application Administration Guide* and the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Overview of Configuration Files

About the SAS Metadata Server Configuration File (omaconfig.xml)

On the machine that hosts the SAS Metadata Server, the **SASMeta/MetadataServer** directory contains a file called omaconfig.xml. This file contains SAS Metadata Server settings such as the following:

- values for the libref and path of the metadata server's repository manager

- the names and locations of the adminUsers.txt and trustedUsers.txt files, which contain information about special users of the metadata server

- journaling options

- alert email options

For details about the omaconfig.xml file, see "Reference Information for omaconfig.xml" on page 456.

About Other SAS Server Configuration Files

Server-tier machines contain configuration files for each SAS server. These files specify the SAS system options that are in effect when the server is launched. The following configuration files are present in each server's configuration directory:

sasv9.cfg
: is the primary configuration file for a server. You should not edit this configuration file.

sasv9_usermods.cfg
    enables user modifications to a server configuration. If you need to modify an option or add a new option for a specific server, then edit this file.

logconfig.xml
    specifies the logging configuration for the server. For details, see Chapter 10, "Administering Logging for SAS Servers," on page 131.

The files sasv9.cfg and sasv9_usermods.cfg are also present in the configuration directory for the SAS Application Server (for example, the **SASApp** directory) and in the directory where SAS Foundation is installed. Configuration files are set up to inherit options from one another.

For details about the server configuration files, see "Reference: Configuration Files for SAS Servers" on page 447.

About the Security-Related Configuration Files

On the machine that hosts the SAS Metadata Server, three configuration files that affect security are located in the **SASMeta/MetadataServer** directory:

Note: Be sure to limit access to these files to only those users who require access in order to perform their job functions. In the initial configuration on UNIX and z/OS systems, only the SAS Installer user has access to this directory.

adminUsers.txt
    defines your system's initial unrestricted user. The SAS Deployment Wizard creates this user, which is usually called the SAS Administrator. In the file, the unrestricted user name is preceded by an asterisk. For information about unrestricted users, see "About User Roles for System Administration" on page 19.

trustedUsers.txt
    conveys privileged status to a service account to enable that account to act on behalf of other users. For more information, see "About the Initial User Accounts" on page 22.

trustedPeers.xml
    can be used to limit the scope of trust for connections from SAS sessions to the metadata server. Initially, the scope is not limited. For details, see the "Trusted Peer Connections" topic in the "Authentication Mechanisms" chapter of the *SAS Intelligence Platform: Security Administration Guide*.

About Configuration Files for Client Applications

Configuration files for client applications are described in the following documents:

• *SAS Intelligence Platform: Web Application Administration Guide*

• *SAS Intelligence Platform: Desktop Application Administration Guide*

Overview of Initial Roles, Groups, and Users

About User Roles for System Administration

Overview of System Administration User Roles

A user role is a set of capabilities. Some SAS applications make certain actions available only to users or groups that have a particular role. To enable a user or group to perform those actions, you add the user or group to the appropriate role.

During installation, the SAS Deployment Wizard creates metadata definitions for several user roles. The following initial roles are created for performing system administration tasks:

- Metadata Server: Unrestricted Role

- Metadata Server: User Administration Role

- Metadata Server: Operation Role

- Management Console: Advanced Role

- Management Console: Content Management Role

The SAS Deployment Wizard also creates some additional roles for users of specific client applications, including SAS Enterprise Guide, SAS Web Report Studio, and SAS Add-In for Microsoft Office.

Note:

- To understand how role assignments affect a user's ability to perform the system administration tasks that are documented in this guide, see Chapter 4, "Who Can Do What: Credential Requirements for SAS Management Console Tasks," on page 39.

- For information about how to add users to roles or to define additional roles, see "Managing Users, Groups, and Roles" in the *SAS Intelligence Platform: Security Administration Guide*.

Metadata Server: Unrestricted Role

Note: The initial name of this role is **META: Unrestricted Users Role**, and the initial display name for this role is **Metadata Server: Unrestricted**.

The **Metadata Server: Unrestricted** role has access to all metadata regardless of SAS permissions settings. Users in this role, which are referred to as unrestricted users, can do the following:

- perform all of the functions that users in the **Metadata Server: User Administration** and **Metadata Server: Operation** roles can perform

- access all metadata except user passwords

- continue to access metadata repositories and use features of SAS Management Console when the metadata server is paused to the Administration state

Follow these important guidelines when using an account that is in the **Metadata Server: Unrestricted** role:

- This role is intended only for tasks that require unrestricted access to metadata (for example, adding other users to the **Metadata Server: Unrestricted** role, performing tasks when the metadata server is paused to the Administration state, and creating, deleting, formatting, and unregistering foundation repositories).

- Use the accounts in this role only to log on to SAS Management Console. You should not use these accounts to log on to other client applications.

The SAS Deployment Wizard places one user in this role. This user, which is generally called the SAS Administrator, is specified in the file adminUsers.txt. For details, see "About the Initial User Accounts" on page 22.

Note: The **Metadata Server: Unrestricted** role provides access to the metadata server, not to other SAS servers. Some administration tasks require access to a SAS Application Server, which might require additional credentials. For details, see Chapter 4, "Who Can Do What: Credential Requirements for SAS Management Console Tasks," on page 39.

Metadata Server: User Administration Role

Note: The initial name of this role is **META: User and Group Administrators Role**, and the initial display name for this role is **Metadata Server: User Administration**. The SAS Deployment Wizard assigns the SAS Administrators group to this role.

Users who are assigned to the **Metadata Server: User Administration** role can create and modify users, groups, and roles. Users in this role are authorized to update user passwords. They cannot read existing passwords, except the passwords for their own logins.

For details about the user administration tasks, see the *SAS Intelligence Platform: Security Administration Guide*.

Metadata Server: Operation Role

Note: The initial name of this role is **META: Operators Role**, and the initial display name for this role is **Metadata Server: Operation**. The SAS Deployment Wizard assigns the SAS Administrators group to this role.

Users who are assigned to the **Metadata Server: Operation** role can perform the following tasks:

- stop, pause, resume, and reset (or refresh) the metadata server

- add, delete, format, and unregister metadata repositories (except the foundation repository)

- administer the metadata server backup and recovery facility

Users who perform these tasks must also be assigned to the SAS Management Console Advanced Role, which provides access to the Metadata Manager plug-in.

Management Console: Advanced Role

The **Management Console: Advanced** role is initially configured to allow access to all of the plug-ins in SAS Management Console. The SAS Deployment Wizard assigns the SAS Administrators group to this role.

You must make the following additional role assignments to enable certain functions:

- the **Metadata Server: User Administration** role or the **Metadata Server: Unrestricted** role, to access functionality within the User Manager plug-in

- the **Metadata Server: Operation** role or the **Metadata Server: Unrestricted** role, to access some functions within the Metadata Manager plug-in

Management Console: Content Management Role

The **Management Console: Content Management** role is initially configured to enable access to the following features of SAS Management Console:

- the User Manager, Authorization Manager, and Library Manager plug-ins

- the **Folders** tab

To have access to certain functionality within User Manager plug-in, the user must also be assigned to the **Metadata Server: User Administration**" role or the **Metadata Server: Unrestricted** role.

About the Initial User Groups

User Groups Initially Defined in the Operating System

On some of the machines in your configuration, the following operating system user groups might have been defined during installation:

sas (UNIX only)
> This group is used to control access to the configuration directories on UNIX machines. The group includes the installer (the sas user) and the SAS Spawned Servers account (sassrv). Typically, you will not add any other users to this group.

SAS Server Users (Windows only)
> This group might have been created on Windows machines that have stored process servers, pooled workspace servers, or standard workspace servers installed. During the installation process, you should have assigned this group the right to `Log on as a batch job`, which is required in order to start processes for those servers.
>
> If you are not using Integrated Windows authentication, then you can add users to this group to enable them to start workspace server processes.

SASGRP (z/OS only)
> On z/OS systems, this RACF group is used to control access to the configuration directory. The group is defined with an OMVS segment and is set as the default group for the SAS Installer and SAS Spawned Servers accounts.

User Groups Initially Defined in Metadata

The SAS Deployment Wizard creates the following user groups in metadata. These groups are part of the SAS Intelligence Platform security infrastructure. For information about how they are used to implement security, see the *SAS Intelligence Platform: Security Administration Guide*.

PUBLIC
> a standard group with implicit membership. This group includes everyone who can access the metadata server, either directly or through a trust relationship. A user who does not have an individual identity uses the PUBLIC group identity.

SASUSERS
> a standard group with implicit membership. This group includes all users who have individual identities.

SAS Administrators
> a standard group for metadata administrators. By default, this group is granted broad access to the metadata and has all roles other than the **Metadata Server: Unrestricted** role.

SAS System Services
> a standard group for service identities that need to read server definitions or other system resources.

SAS General Servers
> a standard group whose members can be used for launching stored process servers and pooled workspace servers.

SAS Environment Manager user groups
> standard groups for SAS Environment Manager users. These groups are new with the first maintenance releases for SAS 9.4. The groups include SAS Environment Manager Guests, SAS Environment Manager App Server Tier Users, and SAS Environment Manager Super Users. Users that are members of these groups are mapped to user definitions in SAS Environment Manager with corresponding SAS Environment Manager roles. For more information, see "Controlling Access to SAS Environment Manager" in the *SAS Environment Manager: User's Guide*.

LSF Services
> a group whose members can schedule jobs in the LSF component of Platform Suite for SAS. This group is part of the standard configuration for sites that use Platform Suite for SAS to schedule SAS Web Report Studio reports. The LSF Services group is not needed if you use SAS In-Process Services to schedule reports.
>
> See also:

About the Initial User Accounts

Overview of the Initial User Accounts

During installation, the SAS Deployment Wizard creates several initial user accounts. Some of these user accounts are created for all installations, some of the accounts are optional, and some of the accounts are created only if certain software components are installed. For each account, the following topics provide the default name and user ID, information about whether or when the account is required, the account's purpose and use, and the locations where the account is set up:

- SAS Administrator
- SAS Trusted User
- SAS Installer
- SAS Spawned Servers account
- SAS First User
- SAS Environment Manager Service Account
- SAS Anonymous Web User
- LSF Administrator
- lsfuser

These user accounts might have been assigned different names at your site.

SAS Administrator

Table 2.3 SAS Administrator Characteristics

Type of Installation	Default User Name	Default User ID	Required?	Location of Account
Default settings	SAS Administrator	sasadm@saspw	Yes	Metadata
External authentication selected	SAS Administrator	sasadm	Yes	Metadata and OS

The SAS Administrator user account has privileges that are associated with the **Metadata Server: Unrestricted** role. See "Metadata Server: Unrestricted Role" on page 19. In addition, the SAS Administrator account is initially a member of the SAS Administrators group.

This user is defined in the following locations:

- in the file adminUsers.txt, which is typically located in the following path:

  ***SAS-configuration-directory*/Lev1/SASMeta/MetadataServer**

  This file ensures that your site will always have at least one user with the privileges of an unrestricted user, regardless of what is specified in metadata. You cannot override this user's privileges by modifying the user definition in SAS Management Console.

- in metadata.

- in the operating system of the metadata server machine, if you selected the external authentication option for this user during a custom installation.

In default installations, the SAS Administrator is an internal user account that is known only to SAS and that is authenticated internally in metadata. When internal authentication is used, it is not necessary for this user to have a local or network account.

Note: We recommend that you establish individual metadata administrators rather than sharing the SAS Administrator account. See "Security Tasks" in the *SAS Intelligence Platform: Security Administration Guide*.

SAS Trusted User

Table 2.4 SAS Trusted User Characteristics

Type of Installation	Default User Name	Default User ID	Required?	Location of Account
Default settings	SAS Trusted User	sastrust@saspw	Yes	Metadata
External authentication selected	SAS Trusted User	sastrust	Yes	Metadata and OS

The SAS Trusted User is a privileged service account that can act on behalf of other users on a connection to the metadata server. No user should log on directly as a trusted user, except to perform certain administrative tasks associated with the SAS Information

Delivery Portal. For details about those tasks, see the *SAS Intelligence Platform: Web Application Administration Guide*.

The SAS Trusted User is defined in the following locations:

- in metadata.

- in the file trustedUsers.txt, which is typically located in the following path:

  ***SAS-configuration-directory*/Lev1/SASMeta/MetadataServer**

  A user is granted privileges as a trusted user only if the user is specified in this file.

  Note: Typically, there is no reason to add more IDs to this file. In particular, do not add regular users to this file.

- in the operating system of the metadata server machine, if you selected the external authentication option for this user during a custom installation.

In default installations, the SAS Trusted User is an internal user account that is known only to SAS and that is authenticated internally in metadata. When internal authentication is used, it is not necessary for this user to have a local or network account.

For detailed information about this user, see the *SAS Intelligence Platform: Security Administration Guide*.

SAS Installer

Table 2.5   SAS Installer Characteristics

Default User Name	Default User ID	Required?	Location of Account
SAS Installer	sas	Yes	OS

The SAS Installer is a user account that is used to install and configure SAS software. On UNIX and z/OS systems, this account is the owner of installation and configuration directories and their contents and is the process owner for items such as the metadata server, the OLAP server, and the object spawner. The account should continue to be available after installation so that it can be used to apply maintenance and hot fixes.

The SAS Installer account must be defined in the operating systems of the following machines:

- the metadata server machine

- machines that host an OLAP server

- machines where the object spawner is installed

This user is not defined in metadata.

SAS Spawned Servers Account

Table 2.6   SAS Spawned Servers Account

Default User Name	Default User ID	Required?	Location of Account
SAS Spawned Servers	sassrv	Yes	OS and metadata (as a login for the SAS General Servers group)

The SAS Spawned Servers account is the initially configured process owner for pooled workspace servers and stored process servers. An account for this user must be defined in the operating system of the following machines:

• machines that host a stored process server

• machines that host a pooled workspace server

During the installation process on Windows machines, this user should have been assigned the right to **Log on as a batch job**. This right can also be assigned by adding the user to the SAS Server Users group.

This user does not have an individual metadata identity. However, a login for this user is defined for the SAS General Servers group.

SAS Environment Manager Service Account

Table 2.7   SAS Environment Manager Service Account Characteristics

Default User Name	Default User ID	Required?	Location of Account
SAS Environment Manager Service Account	sasevs@saspw	Yes	Metadata

Effective with SAS 9.4M1, the SAS Environment Manager Service Account is required for communications between the SAS Environment Manager agent and the SAS Environment Manager server. The account also enables SAS Environment Manager plug-ins to access the SAS Metadata Server.

This account is an internal user account that is known only to SAS and that is authenticated internally in metadata. The account has privileges that are associated with the **Metadata Server: Unrestricted** role and is initially a member of the SAS Administrators group and the SAS Environment Manager Guests group on page 22 .

SAS First User

Table 2.8   SAS First User Characteristics

Default User Name	Default User ID	Required?	Location of Account
SAS Demo User	sasdemo	No	Metadata and OS

The SAS First User is an optional account that can serve as a generic end user when you are testing any of the SAS client applications. During installation, the Software Deployment Wizard enables you to specify whether to create this user.

If you selected the option to create this user, then the user's account is defined in the following locations:

- in metadata

- in the operating system of the metadata server machine and workspace server machine

During the installation process on Windows machines, this user should have been assigned the right to **Log on as a batch job**. This right can also be assigned by adding the user to the SAS Server Users group.

SAS Anonymous Web User

Table 2.9   SAS Anonymous Web User Characteristics

Type of Installation	Default User Name	Default User ID	Required?	Location of Account
Default settings	SAS Anonymous Web User	webanon@saspw	No	Metadata
External authentication selected	SAS Anonymous Web User	webanon	No	Metadata and OS

The SAS Anonymous Web User is an optional account that is used to grant clients access to applicable SAS Web Infrastructure Platform components. When web clients request access to web services, they are not prompted for credentials but instead are granted access under this user account.

This user is defined in the following locations:

- in metadata. In default installations, the SAS Anonymous Web User is an internal user account that is known only to SAS and that is authenticated internally in metadata. When internal authentication is used, it is not necessary for this user to have a local or network account.

- in the operating system of the metadata server machine, only if you selected the external authentication option for this user during a custom installation.

LSF Administrator

Table 2.10   LSF Administrator Characteristics

Default User ID	Required?	Location of Account
none	Yes, if Platform Suite for SAS is installed	OS

The LSF administrator is the primary administrator for the Platform scheduling server and the owner of the Process Manager server. This user is required only if you have installed Platform Suite for SAS in support of either scheduling or grid computing.

The LSF administrator account must be defined in the operating system of the machine where Platform Suite for SAS is installed. This user must have full control of the LSF and Process Manager directories. On Windows systems, this user must belong to the Administrators Group and must have rights to `Act as part of the operating system` and `Log on as a batch job`.

This user is not defined in metadata.

lsfuser

Table 2.11 *lsfuser*

Default User ID	Required?	Location of Account
lsfuser	Yes, if Platform Suite for SAS is installed and is used to schedule SAS Web Report Studio reports	OS, metadata (as a login for the LSF Services group), and password file in LSF

The lsfuser account is used by default when you schedule SAS Web Report Studio reports using the LSF component of Platform Suite for SAS. The lsfuser account must be defined in the operating system of the machine where Platform Suite for SAS is installed. On Windows machines, the account must also be added to the password file in the LSF software. For details, see "Enabling Report Scheduling With Platform Suite for SAS" in *Scheduling in SAS*.

This user does not have an individual metadata identity. However, a login for this user is defined for the LSF Services group.

The lsfuser account is not needed if you use SAS In-process Services to schedule reports.

Chapter 3

Overview of the Administration Tools

Overview of SAS Management Console

About SAS Management Console

As you administer the SAS Intelligence Platform, one of your primary tools will be SAS Management Console. SAS Management Console is a Java application that provides a single point of control for administering your SAS servers and for managing metadata objects that are used throughout the SAS Intelligence Platform.

Whenever the SAS Metadata Server is running, you can use SAS Management Console to connect to the SAS Metadata Server and view and manage the metadata objects that are stored in the server's metadata repositories. The SAS Management Console user interface includes a **Plug-ins** tab, a **Folders** tab, and a **Search** tab that you can use to access and manage metadata, as described in the following topics.

For more information, see "Starting SAS Management Console" in *SAS Intelligence Platform: Desktop Application Administration Guide*.

About SAS Management Console Plug-ins

SAS Management Console is a framework in which a variety of plug-ins are installed. A plug-in is an application module that is designed to create and maintain metadata for a specific type of resource. For example, the Server Manager plug-in is used to create and update metadata to define SAS servers. Some plug-ins include additional functionality.

For example, the Server Manager plug-in enables you to stop, pause, and resume servers and to monitor server activity.

Only certain users can view and use plug-ins. A user's access to plug-ins depends on which roles the user is assigned to and which capabilities are assigned to those roles. See "Overview of Initial Roles, Groups, and Users" on page 19.

If you are authorized to use plug-ins, then the **Plug-ins** tab appears in the left pane when you open SAS Management Console and connect to a metadata server, as shown in this example:
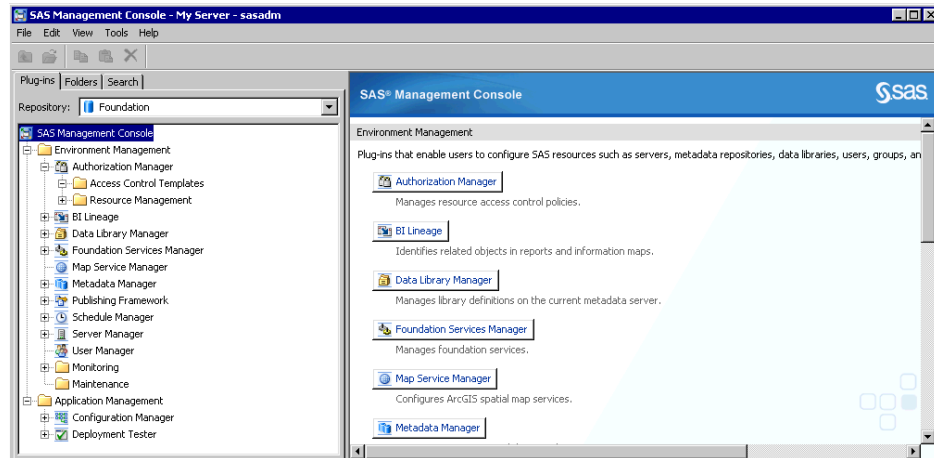
The following table provides the name and purpose of each of the most frequently used SAS Management Console plug-ins. References to related documentation are also provided.

Table 3.1   *SAS Management Console Plug-ins*

Plug-in Name	Plug-in Description
Authorization Manager	Define and maintain access rules to control how users and groups can access metadata definitions. For details, see the *SAS Intelligence Platform: Security Administration Guide*.
BI Lineage	Identify connections between BI objects in the SAS Folders tree. For details, see "Using the BI Lineage Plug-in" on page 250 and the SAS Management Console Help.
Configuration Manager	View and modify configuration attributes for SAS applications including SAS Information Delivery Portal, SAS Web Report Studio, SAS Financial Management, and SAS Strategic Performance Management. For details, see the *SAS Intelligence Platform: Web Application Administration Guide*.
Data Library Manager	Create and maintain definitions for SAS libraries and database schemas. For details, see the *SAS Intelligence Platform: Data Administration Guide*.

Plug-in Name	Plug-in Description
Foundation Services Manager	View and modify deployment configurations for infrastructure and extension services that are used by applications such as SAS Information Delivery Portal, SAS BI Dashboard, and SAS Web Report Studio. For details, see the *SAS Intelligence Platform: Web Application Administration Guide*.
Map Service Manager	Create and maintain map service definitions that link cubes to Esri ArcGIS map services so that Esri Geographic Information System (GIS) spatial map information can be accessed and returned by an OLAP server when the cubes are queried. For details, see the *SAS Intelligence Platform: Web Application Administration Guide*.
Metadata Manager	Perform administration tasks related to the SAS Metadata Server, including the following: • stop, pause, resume, or reset the metadata server • check the status of the metadata server and the metadata repositories • create and manage metadata repositories • back up or restore your metadata repositories • analyze and repair metadata
Publishing Framework	Create and maintain definitions for channels, package subscribers, group subscribers, and event subscribers. For details, see the *SAS Intelligence Platform: Web Application Administration Guide*.
Schedule Manager	Create and maintain schedules for running jobs that are created in SAS Data Integration Studio and SAS Web Report Studio. For details, see *Scheduling in SAS*.
Server Manager	Perform administration tasks related to SAS servers, including the following: • check the status of servers and validate them to ensure that they are configured correctly • stop, pause, quiesce, resume, or refresh the SAS object spawner or the SAS/CONNECT spawner, and stop, pause, resume, or quiesce a process on a SAS OLAP Server, SAS Workspace Server, or SAS Stored Process Server • connect to the metadata server, to components of SAS Application Servers, to the SAS object spawner, or to the SAS/CONNECT spawner to perform monitoring activities Details about the preceding tasks are provided in this guide. You can also use the Server Manager plug-in to view and maintain definitions for servers. For details, see the *SAS Intelligence Platform: Application Server Administration Guide*.
User Manager	Create and maintain definitions for users, groups, and roles. For details, see the *SAS Intelligence Platform: Security Administration Guide*.

Using the SAS Management Console Plug-ins

To use the plug-ins, follow these general steps:

- To display the metadata objects and folders that are associated with a plug-in, click the plus sign next to the plug-in icon. The objects and folders are displayed in a navigation tree.

- To expand a folder or object in the navigation tree, click the plus sign next to it.

- To determine the actions that you can take with an item in the navigation tree (for example, a plug-in, a folder, or an object), right-click the item and select an action from the menu that appears.

- To view the metadata definition for an object in the navigation tree, right-click the object and select **Properties**.

- To obtain help for a plug-in, select the plug-in (or any object associated with the plug-in) in the navigation tree, and select **Help** ⇨ **Help on <plug-in name>**.

Some of the plug-ins have wizards that assist you in creating new metadata objects. For example, Data Library Manager has a New Library Wizard that helps you create a new SAS library. To access the wizard, you expand the **Data Library Manager** node, right-click the **SAS Libraries** folder, and select **New Library**.

For detailed information about using SAS Management Console and its plug-ins, see the product Help and the applicable administration guides.

About the SAS Management Console Folders Tab

The **Folders** tab in SAS Management Console displays the hierarchy of SAS folders. These folders are used to store metadata for business intelligence content such as libraries, tables, jobs, cubes, information maps, and reports. The folders also store SAS system content that is not directly accessed by business users.

As a system administrator, you will use this view for tasks such as the following:

- setting up a folder structure for your users

- setting permissions on folders and objects

- importing and exporting metadata and associated files (for example, when promoting content to production)

For a newly installed system, the folder organization is similar to the following:
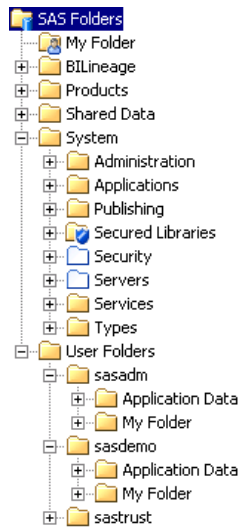
For details, see "About the SAS Folder Structure" on page 243.

Overview of SAS Environment Manager

SAS Environment Manager, which is new with SAS 9.4, is a web-based administration solution for a SAS environment. The application enables you to perform these tasks:

- administer, monitor, and manage SAS resources, including administering the SAS Web Application Server and monitoring SAS foundation servers

- collect and chart data on metrics for monitored SAS resources, thus creating a comprehensive view of resource health and operation

- monitor log events and reporting alerts

- incorporate the monitoring and managing of IT and SAS resources into a service management strategy by using the detailed metric information stored in the SAS Environment Manager Data Mart and the reports provided in the Report Center

- access modules to perform other aspects of SAS administration (see "About SAS Environment Manager Administration" on page 34)

SAS Environment Manager is based on VMware's Hyperic product, with customizations and plug-ins to optimize the product specifically for a SAS environment. The basic architecture consists of an agent process (running on each middle-tier and server-tier machine) that communicates with a central management server. Agents monitor detected resources and periodically report resource metrics back to the server. The server provides an interface for interacting with those agents, managing the data collected by the agents, distributing plug-ins, creating alerts and escalation procedures based on collected metrics, and graphing the metrics provided through the installed plug-ins.

SAS Environment Manager agents run on all SAS platforms except for z/OS. To access SAS Environment Manager, use your web browser to go to **http://localhost:7080**, where *localhost* is the machine on which the SAS Environment Manager server is installed. After you specify your credentials in SAS Login Manager, the main SAS Environment Manager application appears.

For more information, see *SAS Environment Manager User's Guide* and the Help for SAS Environment Manager, which is available from within the product.

About SAS Environment Manager Administration

SAS Environment Manager Administration is a web application that helps you manage the SAS Intelligence Platform.

- The 2.6 version is available in SAS 9.4M6 (May 2019 update) and later deployments.

- The 2.5 version is available in earlier SAS 9.4 deployments, beginning with SAS 9.4M3 in July 2015.

Comparison of SAS Management Console and SAS Environment Manager

Over the life cycle of SAS 9.4, functions are being added to extend the capabilities of SAS Environment Manager as a centralized administration application for all SAS products. SAS Environment Manager is not currently a complete replacement for SAS Management Console, and no functionality has been removed from SAS Management Console.

Note: This topic describes functionality that is available as of SAS 9.4M6 (May 2019 update). Not all of the features that are listed here are available in prior deployments. See What's New for information about when features were introduced.

SAS Management Console is a desktop application that covers most metadata management and operations tasks. See the following table for tasks that are available only from SAS Management Console:

Table 3.2   Tasks Available Only in SAS Management Console

View server logs and dynamically change logging levels.

Validate servers on the SAS server tier and run the Deployment Tester.

Back up and restore the metadata server, and create and administer metadata repositories.

Schedule flows to run on a scheduling server.

Create and modify metadata definitions for other types of SAS libraries and servers.

Create and modify metadata definitions for database schemas, map services, servers, stored processes, publication channels, and subscribers.

Display lineage information.

Promote (export and import), copy, and paste metadata.

View and modify configuration attributes for SAS applications, and view and modify deployment configurations for infrastructure and extension services that are used by these applications.

Update metadata functionality.

SAS Environment Manager Administration is available from SAS Environment Manager. See the following table for tasks that are available only from SAS Environment Manager Administration:

Table 3.3   *Tasks Available Only in SAS Environment Manager Administration*

Administration Task
Manage mobile devices.
Manage and monitor LASR servers.[*]
Manage and monitor HDFS folders and tables.[*]
Manage LASR metric alert definitions that are created in SAS Visual Analytics.[*]

[*]   This functionality is available only if SAS Visual Analytics 7.5 is deployed.

SAS Environment Manager is a web-based administration solution for a SAS environment. The type of functionality that is available from SAS Environment Manager includes the following:

- Start, stop, and restart the SAS Web Application Server; and start, stop, and reload web applications.

- View metrics on the availability, performance, utilization, resource consumption, and throughput of server machines on the middle tier and the SAS server tier. Set up alerts based on these metrics.

- Use reporting tools to obtain a comprehensive view of the performance and status of your SAS environment and its resources.

- Start servers on the SAS server tier and middle tier.

Overview of SAS Deployment Manager

SAS Deployment Manager is a graphical user interface that enables you to do the following:

- update passwords for the service accounts that were configured when you ran the SAS Deployment Wizard. For details, see "Managed Passwords" in *SAS Intelligence Platform: Security Administration Guide*.

- rebuild and redeploy web applications that have previously been configured but whose configuration has changed. For details, see "Administer SAS Web Applications" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

- uninstall SAS software from the local machine. For details, see "Uninstalling Your SAS Software" in *SAS Intelligence Platform: Installation and Configuration Guide*.

- remove one or more components of a SAS Intelligence Platform configuration from your environment. For details, see "Removing a SAS Configuration" in *SAS Intelligence Platform: Installation and Configuration Guide*.

- update setinit (license) information in metadata for some SAS solutions that depend on a SAS middle tier. For details, see *Software License Renewal Instructions—SAS for Microsoft Windows for x64 Planned Deployment* and *Software License Renewal Instructions—SAS for UNIX Environments Planned Deployment*, which are available at http://support.sas.com/documentation/installcenter/94/index.html.

- manage the default associations between file types and SAS software. For details, see the *SAS Deployment Wizard and SAS Deployment Manager: User's Guide*.

- change the host names (including the network domains to which they belong) of server machines in your deployment. For details, see Chapter 33, "Using the SAS Deployment Manager to Update Host Name References," on page 475.

- apply downloaded hot fixes to your SAS software. For details, see the *SAS Guide to Software Updates and Product Changes*.

- change the passphrase that is used to encrypt stored passwords. For details, see "How to Re-encrypt Stored Passwords" in *SAS Intelligence Platform: Security Administration Guide*.

- configure the language and region for SAS Foundation and certain SAS applications. For details, see "Change Locale for SAS" in *SAS Intelligence Platform: Installation and Configuration Guide*.

- configure and manage the SAS Deployment Agent service. For details, see the *SAS Deployment Wizard and SAS Deployment Manager: User's Guide*.

Overview of SAS Web Administration Console

SAS Web Administration Console is a web-based interface that enables you to do the following:

- monitor which users are logged on to SAS web applications

- view audit reports of logon and logoff activity

- manage notification templates and letterheads

- manage web-layer authorization, including privileges, roles, and permissions

- access the SAS Content Server Administration Console in order to manage folders and permissions for the SAS Content Server

- view the current configuration of web applications

- dynamically adjust logging levels for some web applications

For details, see "Use the SAS Web Administration Console " in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Chapter 4

Who Can Do What: Credential Requirements for SAS Management Console Tasks

About Security for System Administration Tasks

The system administration tasks that can be performed using SAS Management Console affect the stability of your servers, the integrity of your data and metadata, and the performance of client applications. Therefore, it is important to limit access to system administration tasks to only those users who:

- have the necessary skills, background, and knowledge

- need to perform system administration tasks in order to fulfill their job functions

The security model for the SAS Intelligence Platform model enables you to control access to system administration tasks by adding users to administrative roles, adding users to groups, and granting permissions on metadata. If necessary, you can also create new roles.

This chapter describes the group memberships, role memberships, and other criteria that are required for users to perform the system administration tasks that are documented in this guide. Requirements are provided for two different approaches to system administration:

centralized administration
    means that a single administrative user (or a small group of users) performs all administrative functions for your SAS deployment, with no division of responsibility

delegated administration
    means that various types of administrative functions for your SAS deployment are divided among different administrative users or groups of users

For some functions, the requirements are the same regardless of whether you are using centralized or delegated administration.

The requirements in this chapter reflect initial settings in a default installation. Role definitions and group membership might be configured differently at your site. For an in-depth description of the security model, or for information about modifying the security structure, see the *SAS Intelligence Platform: Security Administration Guide*.

Note: This chapter addresses security only for administration tasks that are performed in SAS Management Console. For SAS Environment Manager security information, see "Controlling Access to SAS Environment Manager" in the *SAS Environment Manager: User's Guide*.

Who Can Do What in the Metadata Manager Plug-in

Use the following table to determine the group memberships, role memberships, and other criteria that enable users to perform tasks in the Metadata Manager plug-in in SAS Management Console:

Table 4.1 *Who Can Do What in the Metadata Manager Plug-in*

Actions	Requirements[*]	
	Centralized Administration Approach	**Delegated Administration Approach**
Check the access state of the metadata server, metadata repositories, and repository manager	Membership in the SAS Administrators group	Membership in the **Management Console: Advanced** role.
Create, register, and delete custom repositories Change a repository's properties, including the access mode, the location of users' folders, and mapping for notes and documents	Membership in the SAS Administrators group	Membership in the **Management Console: Advanced** and **Metadata Server: Operation** roles, and WriteMetadata permission on the Root folder of the SAS Folders tree.
Stop, pause, or resume the metadata server Administer metadata server backups Test metadata server alert e-mails	Membership in the SAS Administrators group	Membership in the **Management Console: Advanced** and **Metadata Server: Operation** roles.
Continue to use all functions of SAS Management Console when the metadata server is paused to the Administration state Analyze, repair, or upgrade metadata Initialize or unregister a custom repository	Membership in the **Metadata Server: Unrestricted** role	

	Requirements[*]	
Actions	**Centralized Administration Approach**	**Delegated Administration Approach**
Add, initialize, register, unregister, and delete a foundation repository.[**]	Membership in the SAS Administrators group, specification as an unrestricted user in adminUsers.txt, and an external user ID that can be authenticated on the metadata server host machine	

[*] These columns list minimum requirements. The **Metadata Server: Unrestricted** role can be substituted for any of the roles and groups that are specified here.

[**] Deleting or unregistering the foundation repository makes the metadata server unusable. These actions should be taken only in special situations.

As a best practice, users who are members of the SAS Administrators group should be defined as internal users (rather than external users) in metadata. For more information, and to understand what role memberships and permissions are derived from membership in this group, see "About Adding Users to the SAS Administrators Group" on page 42.

Who Can Do What in the Server Manager Plug-in

Use the following table to determine the group memberships, role memberships, and other criteria that enable users to perform tasks in the Server Manager plug-in in SAS Management Console:

Table 4.2   Who Can Do What in the Server Manager Plug-in

	Requirements[*]	
Actions	**Centralized Administration Approach**	**Delegated Administration Approach**
Create or update server metadata Validate servers Connect to servers and spawners and view the following tabs[**]: • Metadata server **Connections** and **Options** tabs • Object spawner **Options** tab • OLAP server **Connections**, **Sessions**, and **Options** tabs	Membership in the SAS Administrators group Validation of a standard (non-pooled) workspace server also requires external credentials.	Membership in the **Management Console: Advanced** role Validation of a standard (nonpooled) workspace server also requires external credentials .
Connect to servers and spawners (except the metadata server) and view all tabs[**] Adjust the logging level for a server, spawner, or server process (except the metadata server) Stop, pause, resume, or quiesce a server, spawner, or server process (except the metadata server) Refresh the object spawner	Membership in the SAS Administrators group	Membership in the **Management Console: Advanced** role, and Administer permission on server metadata

	Requirements[*]	
Actions	**Centralized Administration Approach**	**Delegated Administration Approach**
Connect to the metadata server and view the **Clients**, **Loggers**, and **Logs** tabs[**] Adjust the metadata server's logging levels Stop, pause, or resume the metadata server	Membership in the SAS Administrators group	Membership in the **Management Console: Advanced** and **Metadata Server: Operation** roles

  *   These columns list minimum requirements. The **Metadata Server: Unrestricted** role can be substituted for any of the roles and groups that are specified here.

**   To view tabs, you must right-click on the server or spawner and click **Connect**.

As a best practice, users who are members of the SAS Administrators group should be defined as internal users (rather than external users) in metadata.

For more information, and to understand what role memberships and permissions are derived from membership in this group, see "About Adding Users to the SAS Administrators Group" on page 42.

About Adding Users to the SAS Administrators Group

When you add a user to the SAS Administrators group, the user derives the following memberships and permissions (if the group's initial settings have not been changed):

- membership in the **Management Console: Advanced** role

- membership in the **Metadata Server: Operation** role

- membership in the **Metadata Server: User Administration** role. This role is required in order to perform most actions in the User Manager plug-in.

- WriteMetadata permission on the Root folder of the SAS Folders tree

- Administer permission on server metadata

- ReadMetadata on server metadata in the SASMeta context

If you use a centralized (rather than delegated) approach to system administration, then the SAS Administrators group provides a convenient mechanism for enabling users to perform administrative tasks. Centralized administration means that a single administrative user (or a small group of users) performs all administrative functions for your SAS deployment, with no division of responsibility.

As a best practice, users who are members of the SAS Administrators group should be defined as internal users (rather than external users) in metadata whenever possible. For details about creating administrative users, see "Add Administrators" in *SAS Intelligence Platform: Security Administration Guide*.

Requirements for Accessing a Standard Workspace Server

Special rights are required for tasks that involve running processes on a standard (non-pooled) workspace server. In SAS Management Console, these tasks include the following:

- validating a workspace server

- using the Import SAS Package and Export SAS Package wizards or the batch promotion tools to import or export physical content that is associated with stored processes, tables, jobs, libraries, or external files

The requirements for performing these tasks depend on which authentication method is used and which server is being accessed:

- If you use host (credential-based) authentication, the following requirements must be met:

  - The user's account must be known to the workspace server host. On Windows, the account must have rights to **Log on as a batch job** on the host machine. Typically, you would add the user to an operating system group that has this right (for example, the SAS Server Users group).

  - The user's account must correspond to a metadata identity that has the ReadMetadata permission for the server definition.

- If you use Integrated Windows authentication (IWA), the following requirements must be met:

  - The user's account must be known to the workspace server host. It is not necessary to have rights to **Log on as a batch job**.

  - The user's account must correspond to a metadata identity that has the ReadMetadata permission for the server definition.

  - Integrated Windows authentication must be selected in the connection profile that the user uses to log on to SAS Management Console. In addition, Integrated Windows authentication must be fully configured on the workspace server.

- If the workspace server is configured to use SAS token authentication, then no credentials on the workspace server host are necessary. The user can log on to SAS Management Console with either an internal account or an external account. The account must correspond to a metadata identity that has the ReadMetadata permission for the server definition.

For detailed information about configuring and using the authentication methods for SAS servers, see the *SAS Intelligence Platform: Security Administration Guide*.

Clearing the Credentials Cache

If a user attempts to access a server in SAS Management Console but does not have the required credentials, the user is prompted for a user ID and password. The entries are stored in the credentials cache, and they continue to be used for the duration of the SAS Management Console session. If the user later wants to use different credentials to

access a server, the user can clear the credentials cache by clicking **File ⇨ Clear Credentials Cache**.

Clearing the Object Spawner's Decision Cache

If the object spawner denies access to a server because of the lack of ReadMetadata permissions, and the user is then granted the needed permission, you must reset the object spawner's decision cache before the user tries to connect again. The decision cache is the location in which the spawner's credentials decisions are stored. To clear this cache, expand the Server Manager tree in SAS Management Console. Next, expand the object spawner, right-click the host node, and click **Connect**. After you have connected, right-click the host node again, and click **Clear Spawner Credentials Cache**.

Chapter 5

What to Do Next:
Administration Tasks

Overview of Administration Tasks

Before you perform the administration tasks in this chapter, you should have
successfully completed the following tasks:

- Use the SAS Deployment Wizard to install and configure your software on each machine in your environment.

- Complete the post-installation tasks that are described in your Instructions.html file, which is located in the **Documents** subdirectory of your configuration directory. These tasks include performing manual configuration steps (if applicable), validating your SAS servers, validating your web applications, and backing up your system.

After the SAS Intelligence Platform has been installed, configured, and validated, you will have a functional system. You can begin performing administration tasks that are necessary to fully implement SAS Intelligence Platform in your environment. These include the following tasks:

- "First-Priority Setup Tasks" are necessary to protect the integrity of your system. You should perform these tasks first, as soon as possible after you have completed your installation.

- "Standard Setup Tasks" enable the users in your organization to begin using SAS Intelligence Platform client applications to access and analyze your data.

- "Optional Setup Tasks" address specific requirements in your environment.

- "Ongoing System Administration Tasks" keep the SAS Intelligence Platform operational.

First-Priority Setup Tasks

Summary of First-Priority Setup Tasks

The following tasks are necessary to protect the integrity of your system. Complete these steps as soon as possible after installation, before you complete any of the other tasks that are outlined in this chapter.

Table 5.1 *First-Priority Setup Tasks*

Task	Description
Check for and apply SAS hot fixes.	A best practice is to check for and apply SAS hot fixes immediately after deployment and before putting your SAS system into production. Doing so ensures that you have the latest important SAS hot fixes relevant to your site.
	For more information, see SAS 9.4 Guide to Software Updates and Product Changes.
Check for and apply SAS security updates.	A best practice is to check for and apply SAS security updates immediately after deployment and before putting your SAS system into production. Doing so ensures that you have a secure site. See Security Bulletins for more information.

Task	Description
Secure the SAS configuration on each server machine.	For a secure deployment, the configuration directory on each server machine must be protected by operating system controls. These controls prevent inappropriate access to repository data sets, server scripts, server logs, and configuration files.
	On Windows systems, all configuration directories, files, and scripts are owned by the user who performs the installation. You must update the permissions as shown in "Recommended Operating System Protections for Windows Machines". These recommendations assume that your SAS servers and spawners run as services under the Local System account.
	On UNIX and z/OS systems, the SAS Deployment Wizard automatically applies the appropriate permissions. The default permissions are shown in "Default Operating System Protections for UNIX and z/OS Machines".
Establish a formal, regularly scheduled backup process.	Establish a formal, regularly scheduled backup process that includes your metadata repositories as well as the associated physical files.
	SAS includes a server-based facility that performs SAS Metadata Server backups automatically on a scheduled basis. By default, these backups are scheduled to run at 1:00 a.m. every day except Sunday. As a best practice, you should modify your backup configuration to specify a storage device other than the device that is used to store the metadata repositories and the server configuration files. Be sure to include this backup location in your regular system backups. See Chapter 12, "Backing Up and Recovering the SAS Metadata Server".
	It is important to back up the physical data that is associated with the metadata so that related information is synchronized if a restore becomes necessary. The Deployment Backup and Recovery tool, new with SAS 9.4, provides an integrated method for backing up and recovering SAS content across multiple tiers and machines. The backup includes the SAS Metadata Server and resources such as the SAS Content Server and server configuration files. If this tool has been configured, scheduled backups are run each Sunday at 1:00 a.m. by default. See Chapter 28, "Using the Deployment Backup and Recovery Tool".
	For guidance in setting up a backup process, see:
	• Chapter 11, "About Backing Up and Restoring Your SAS Content"
	• "Setting Up Your Backups"

Recommended Operating System Protections for Windows Machines

On Windows server machines, you should apply the following operating system protections to your configuration directory. All of these directories are located in **SAS-configuration-directory\Lev1**.

Table 5.2 Recommended Operating System Protections on Windows

Directories	Users	Recommended Permissions
SAS-configuration-directory	System and Administrators	Full Control
	All other users	List folder contents, Read

Directories	Users	Recommended Permissions
SAS-configuration-directory\Lev1	System and Administrators	Full Control
	SAS Spawned Servers (sassrv)	On Windows Vista, Windows 7, and Windows Server 2008: Special permissions to Read and Execute[*] On Windows XP: Read and Execute[*]
	Deployment Backup and Recovery Tool user account, also referred to as the backup user (for example, sasbackup)[***]	Read and Write
	All other users	List folder contents, Read
SASMeta\MetadataServer	System and Administrators	Full Control
	SAS Metadata Server: Service Login Account (for example, sassvlgn)[**]	Full Control
		Remove all other users and groups
SASMeta\MetadataServer\Backups (or other SAS Metadata Server backup location)	Deployment Backup and Recovery Tool user account, also referred to as the backup user (for example, sasbackup)[***]	Read
Lev1\SASApp	System and Administrators	Full Control
	SAS Spawned Servers (sassrv)	Windows Vista, Windows 7, and Windows Server 2008 only: Special permissions to Read and Execute[*]
	All other users	List folder contents, Read
Lev1 subdirectories: Documents, ReportBatch, SASMeta, Utilities, Web	System and Administrators	Full Control
	All other users	List folder contents, Read

Directories	Users	Recommended Permissions
`Lev1` subdirectories: • `ConnectSpawner` • `Logs` • `ObjectSpawner` • `SASApp\OLAPServer` • `ShareServer`	System and Administrators	Full Control Remove all other users and groups
`SASApp` subdirectories : `PooledWorkspaceServer`, `StoredProcessServer`	System and Administrators	Full Control
	SAS Spawned Servers (sassrv)	On Windows Vista, Windows 7, and Windows Server 2008 only: Read and Execute, List folder contents, and Read[*]
	All other users	No access
`SASApp` subdirectories: `PooledWorkspaceServer\Logs`, `StoredProcessServer\Logs`	System, Administrators, and SAS Spawned Servers (sassrv)	Full Control[*]
`SASApp` subdirectories: • `ConnectServer\Logs` • `Data\wrsdist` • `Data\wrstemp` • `PooledWorkspaceServer\sasuser` • `StoredProcessServer\sasuser` • `WorkspaceServer\Logs`	System, Administrators, and SAS Spawned Servers (sassrv)	Full Control
sasv9_meta.cfg file	System and Administrators	Read and Write Remove all other users and groups

  * The SAS Deployment Wizard automatically sets these permissions for sassrv.

 ** On Windows, this account is required for SAS Metadata Server clustering on Windows.

*** On Windows, this account is required when using a central vault for the Deployment Backup and Recovery Tool or when using SAS Metadata Server clustering.

Note:

- These recommendations assume that your SAS servers and spawners run as services under the Local System account. If servers and spawners are run under a different account, then grant that account the permissions that are recommended for System.

  For example, if the SAS Metadata Server is run under the SAS Metadata Server: Service Login Account, then grant that account Full Control of `SASMeta\MetadataServer`, as shown in the preceding table.

- If you have configured SAS Metadata Server clustering or middle-tier clustering, be sure to set the appropriate protections on all of the configured nodes.

- You might have selected the custom installation option to place all of your log files in a single directory. If you selected this option, then you need to grant the SAS Spawned Servers (sassrv) user Full Control of the central log destination (for example, **`SAS-configuration-directory\Lev1\Logs`**).

- If users are using SAS Enterprise Guide to create stored processes, then the SAS Spawned Servers (sassrv) account must have Write access to the directory in which stored processes are stored.

- If you enable logging for a SAS Workspace Server, then you need to grant all users of the SAS Workspace Server Full Control of the log directory. (See "Create a Log File for Workspace Server Troubleshooting".)

For information about the configuration directory structure, see "Overview of the Configuration Directory Structure".

Default Operating System Protections for UNIX and z/OS Machines

The following table shows the default operating system protections that are provided automatically for configuration directories on UNIX and z/OS machines. All of these directories are located in **`SAS-configuration-directory/Lev1`**.

Table 5.3 *Default Operating System Protections on UNIX and z/OS*

Directories	Users	Default Permissions
`SAS-configuration-directory`	SAS Installer	Read, Write, and Execute
`SAS-configuration-directory``/Lev1` `Lev1` subdirectories: `Documents`, `ReportBatch`, `SASApp`, `SASMeta`, `Utilities`, `Web`	All other users	Read and Execute
`Lev1` subdirectories: • `ConnectSpawner` • `Logs` • `ObjectSpawner` • `SASApp/OLAPServer` • `SASMeta/MetadataServer` • `ShareServer`	SAS Installer	Read, Write, and Execute
	All other users	No access
`SASApp` subdirectories: `PooledWorkspaceServer`, `StoredProcessServer`	SAS Installer	Read, Write, and Execute
	sas group	Read and Execute

Directories	Users	Default Permissions
SASApp subdirectories: • **ConnectServer/Logs** • **Data/wrsdist** • **Data/wrstemp** • **PooledWorkspaceServer/Logs** • **PooledWorkspaceServer/sasuser** • **StoredProcessServer/Logs** • **StoredProcessServer/sasuser** • **WorkspaceServer/Logs**	SAS Installer	Read, Write, and Execute
	sas group	Read, Write, and Execute
sasv9_meta.cfg file	SAS Installer	Read and Write
	All other users	no access

Note:

- Make sure that the SAS Spawned Servers account (sassrv) is a member of the sas group, which has the necessary permissions to server configuration files and log directories.

- You might have selected the custom installation option to place all of your log files in a single directory. If you selected this option, then you need to grant either the sas group or the SAS Spawned Servers (sassrv) user Read, Write, and Execute permission on the central log destination (for example, *SAS-configuration-directory*/Lev1/Logs).

- If users are using SAS Enterprise Guide to create stored processes, then the SAS Spawned Servers (sassrv) account must have Write access to the directory in which stored processes are stored.

- If you enable logging for a SAS Workspace Server, then you need to grant all users of the SAS Workspace Server Read, Write, and Execute permission on the log directory. (See "Create a Log File for Workspace Server Troubleshooting").

- For information about the configuration directory structure, see "Overview of the Configuration Directory Structure".

Standard Setup Tasks

The following administration tasks are essential to enable the users in your organization to begin using SAS Intelligence Platform client applications to access and analyze your data.

Table 5.4 *Standard Setup Tasks*

Task	Description
Add users and manage access.	Standard security setup activities include the following: • creating SAS administrators and regular SAS users • managing access to metadata, data, and application functionality For more information, see "Selected Tasks" in *SAS Intelligence Platform: Security Administration Guide*.
Establish connectivity to your data sources.	To enable the client applications in the SAS Intelligence Platform to access your data sources (including SAS data sets, third-party relational databases, and data from Enterprise Resource Planning systems), you must create metadata objects that represent your data. For example, to enable users to access data in a library of SAS data sets, you must define metadata objects that represent the SAS library and each of the tables in the library. For more information, see Chapter 4, "Connecting to Common Data Sources".
Set up your metadata folder structure.	SAS Intelligence Platform clients use a hierarchy of SAS folders to store metadata for content such as libraries, tables, OLAP schemas, jobs, information maps, and reports. The initial structure provides private folders for individual users and a separate area for shared data. Within these folders, you should create a customized folder structure that meets your specific needs. For more information, see Chapter 18, "Working with SAS Folders". *Note:* Be sure to secure access to the folders as described in the *SAS Intelligence Platform: Security Administration Guide*.

Optional Setup Tasks

It might be necessary for you to modify your initial configuration to meet specific requirements in your environment. Optional administration and configuration tasks include the following:

Table 5.5 *Optional Setup Tasks*

Task	Description
Install sas.servers as a boot script.	On UNIX machines, you can choose to install the sas.servers script as a boot script so that the SAS servers start automatically when you start the machine. The comments in the sas.servers script contain installation instructions that are specific to your operating environment. For more information, see "Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers".
Optimize the performance of the SAS Metadata Server.	To optimize the performance of the SAS Metadata Server, you might want to adjust the maximum number of threads used by the server. See "Configuring the Number of Threads Used by the Metadata Server". For other performance-related tasks, see Chapter 15, "Managing Metadata Server Memory".

Task	Description
Modify the configuration of your processing servers.	Here are some server configuration changes that you might want to make for performance optimization or other reasons: • Define a cluster of load-balanced SAS Workspace Servers in order to balance a load across SAS Workspace Servers that you have installed on multiple hosts. This type of configuration is most useful when you have a large number of users (such as data integration specialists) using a SAS Workspace Server for relatively long-running jobs. For more information, see "Understanding Server Load Balancing" in *SAS Intelligence Platform: Application Server Administration Guide*. • Modify load balancing or algorithms for SAS Stored Process Servers and SAS Pooled Workspace Servers. For more information, see "Understanding the Load Balancing Algorithms" in *SAS Intelligence Platform: Application Server Administration Guide*. • If your SAS server metadata contains characters other than those typically found in the English language, then you must start your SAS server with an ENCODING= or LOCALE= system option that accommodates those characters. For more information, see "Encoding and Locale Information" in the *SAS Intelligence Platform: Application Server Administration Guide*. • If you are submitting large jobs (for example, JAVA GRAPH jobs) on z/OS servers, then you might need to specify a larger region size. For more information, see "Managing Memory" in the *SAS Companion for z/OS*. • If you want your BI output to always use the time zone of the server's operating system, then you can specify the TIMEZONE system option in a restricted configuration file on the server tier. This option prevents users from specifying different time zones. For more information, see "TIMEZONE= System Option" and "Restricted Options" in the *SAS System Options: Reference*.
Optimize web application performance.	For information about configuring middle-tier components for better efficiency and performance, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*.
Adjust server logging.	You can change your server configurations to obtain additional logging information for troubleshooting, performance management, or other purposes. For more information, see Chapter 10, "Administering Logging for SAS Servers" and *SAS Logging: Configuration and Programming Reference*.
Enable job and report scheduling.	To enable SAS scheduling features, you can install and configure scheduling software, configure operating system scheduling, or configure in-process scheduling. For more information, see *Scheduling in SAS*.
Increase Java heap memory allocation for desktop applications.	To process large amounts of data, you might need to modify the Java heap memory allocation that is configured for SAS Management Console, SAS Data Integration Studio, SAS OLAP Cube Studio, SAS Information Map Studio, and SAS IT Resource Management. See "Managing the Java Heap Used by Desktop Applications" in *SAS Intelligence Platform: Desktop Application Administration Guide*.
Set up change management for SAS Data Integration Studio users.	If you want to implement change management for SAS Data Integration Studio, then you must set up project repositories and at least one change-managed folder. For more information, see "Administering SAS Data Integration Studio" in *SAS Intelligence Platform: Desktop Application Administration Guide*.
Collect ARM log information for SAS Data Integration Studio batch jobs.	If you want to collect ARM log information for SAS Data Integration Studio jobs that are run on a batch basis, then you must enable logging for the batch server that executes the jobs. For more information, see "Collecting ARM Log Information for SAS Data Integration Studio Jobs".

Ongoing System Administration Tasks

The following table lists important system administration tasks that you need to perform on an ongoing basis to keep the SAS Intelligence Platform operational.

Table 5.6 *Ongoing System Administration Tasks*

Task	Documentation Reference
Start, stop, pause, resume, and refresh the servers that are used in the system.	Chapter 6, "Operating Your Servers"
Check the status of a server of or of a metadata repository.	Chapter 7, "Checking the Status of Servers"
Monitor the activity of servers.	Chapter 9, "Monitoring the Activity of SAS Servers"
Perform regular full backups of the SAS Metadata Server.	Chapter 11, "About Backing Up and Restoring Your SAS Content"
Use server logs and configure logging options.	Chapter 10, "Administering Logging for SAS Servers" and *SAS Logging: Configuration and Programming Reference*
Create a new metadata repository.	Chapter 17, "Managing SAS Metadata Repositories"
Promote individual metadata objects or groups of objects.	Chapter 20, "Promotion Tools Overview"

Note: The preceding table includes only the tasks that are documented in this guide. Other important administration tasks are described in the following documents:

- *SAS Intelligence Platform: Security Administration Guide*

  You can use "Checklist for a More Secure Deployment" in *SAS Intelligence Platform: Security Administration Guide* to ensure that your environment remains secure on an ongoing basis.

- *SAS Intelligence Platform: Data Administration Guide*

- *SAS Intelligence Platform: Application Server Administration Guide*

- *SAS Intelligence Platform: Desktop Application Administration Guide*

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

- *SAS Intelligence Platform: Web Application Administration Guide*

- *Scheduling in SAS*

- *SAS/SHARE User's Guide*

- *SAS/CONNECT User's Guide*

You can access these documents.

Best Practices for Ensuring the Integrity of Your System

Keep Necessary Documentation Available

SAS administrators should keep the following documentation available for each environment:

- the operational document specific to your environment to assist in transferring information to new administrators or for vacation coverage

- an updated store of user IDs and passwords

- a complete log of changes and events that take place in the SAS environment. This log is helpful if issues are encountered. The following changes should be included in the log:

  - host name changes

  - updates

  - patches

Start and Stop SAS Servers in the Correct Order

It is critical that the SAS servers be started and stopped in the recommended order as documented in "Overview of Server Operation". The SAS_lsm tool described in Usage Note 58231: Utility that manages multi-tiered SAS services for Unix/Linux deployments is one of the options to ensure that this recommendation is followed consistently.

Back Up Your Environment

Maintain a regular schedule of deployment backups, as described in Chapter 28, "Using the Deployment Backup and Recovery Tool".

Use operating system commands to back up your file system contents on a regular basis when server processes are stopped. Verify that your backup process is successful. Verify that you can restore from your backup.

Maintain a daily schedule of SAS Metadata Server backups, as described in Chapter 12, "Backing Up and Recovering the SAS Metadata Server".

By default, the SAS Metadata Server retains backups for 7 days. If you have enough available storage in the backup location, it is recommended that you increase that setting to retain at least 30 days of metadata backups. This ensures that a valid backup is available if a metadata corruption issue is encountered. If there is insufficient storage available in the backup location to support this, it is possible to store metadata backups at an alternative location. You should perform a file system backup of the metadata backup location nightly to allow for the recovery of older backups that were removed by the metadata server in accordance with your file system backup retention policy.

Test Alert Email Configuration

SAS Metadata Server automatically sends an email to the designated email address when specific errors occur. This enables the SAS administrator to take timely action to correct the issue.

To test the alert email:

1. Log on to SAS Management Console.

2. Expand **Metadata Manager**, and right-click the active server.

3. Select **Properties**, and click **Send Test Message**.

4. The settings for the alert email are in the omaconfig.xml and sasv9.cfg files that are located in the **sas-config-dir/SASMeta/MetadataServer** directory.

For more information, see Backup Alert Notifications in *SAS Environment Manager 2.5 Administration: User's Guide*.

Regularly Assess Your Environment for Available Hot Fixes

SAS makes hot fixes available for the customer to download and install. For more information, see the following resources:

- SAS Hot Fix Analysis, Download and Deployment Tool

- SAS Hot Fix Announcements Communities Page

- How to learn about hot fixes to SAS software

Update SAS Licenses and SSL Certificates

SAS licenses and SSL certificates must be updated before the expiration date.

Monitor Your Environment for System Resource Limitations

Based on the system resource limitations defined in the system requirements documentation, monitor your environment for available disk space, CPU, I/O, file descriptors, ulimits, and memory. Use system tools and the Environment Manager provided by SAS to monitor your server health and issue alerts as described in "Using SAS Environment Manager to Monitor SAS Servers".

For Linux, see System Requirements for SAS 9.4 Foundation for Linux for x64.

For Windows, see System Requirements for SAS 9.4 Foundation for Microsoft Windows.

Perform Regular Metadata Maintenance

Perform regular maintenance of metadata as described in "Running the Metadata Analyze and Repair Tools".

Long-Running SAS Sessions

To monitor long-running SAS sessions:

1. Check for long-running SAS sessions that might be abandoned and are no longer connected to a user session or a batch process.

2. Check SAS Enterprise Guide sessions that are left running overnight by users. These can lock the SAS tables.

These sessions might interfere with the nightly backup processes. Both of these tasks can be successfully scripted. Your SAS consultants can help you find and adapt such scripts in your environment.

Clean Up Abandoned SASWork Directories

The SAS Cleanwork Utility for UNIX or Windows can be used for this task.

- For the UNIX environment, see "SAS Usage Utilities: cleanwork" in *SAS Companion for UNIX Environments*.

- For the Windows environment, see "Cleanwork Utility" in *SAS Companion for Windows*.

Run Scripts to Archive Log Files

Compress and archive log files periodically to save disk space. If you have servers that need to be restarted in order to rotate their logs, schedule a daily task to restart each of them. Log files are essential for troubleshooting issues, so retain the archived log files for a period of time to ensure that they are not needed by SAS Technical Support.

SAS Visual Analytics Monitoring

If you have installed SAS Visual Analytics on your machine, monitor the Autoload feature to ensure that it is periodically refreshing the tables when SAS Visual Analytics is running.

Monitor the audit reports to review information about the usage of your SAS Visual Analytics environment.

SAS Grid Monitoring

If you have a grid environment, monitor the distribution of the interactive and batch workload to ensure that the health of your grid is good. For further guidance, see SAS Grid Manager.

Maintenance of the SAS Shared Services Database

For more information about the SAS Shared Services database, see the following resources:

- Tuning the PostgreSQL Data Server in *SAS 9.4 Web Applications: Tuning for Performance and Scalability*

- Monitoring Database Usage with SAS Environment Manager

Maintenance of the PostgreSQL Database

To perform regular maintenance on the PostgreSQL database, see "Tuning the PostgreSQL Data Server" in *SAS Web Applications: Tuning for Performance and Scalability*.

Allow Only the SAS Metadata Server to Access Your Metadata Repositories

The **MetadataRepositories** and **rposmgr** subdirectories, which are located in the SAS configuration directory, are critical to the operation of your system. These data sets contain metadata that defines your servers, users, access levels, enterprise data sources, and data structures. They also contain metadata for resources that are created and used by SAS applications, including information maps, OLAP cubes, report definitions, stored process definitions, and jobs.

To safeguard the integrity of your system:

• Never move, delete, modify, or directly read the data sets in the **MetadataRepositories** and **rposmgr** directories. These data sets should be accessed only by SAS Metadata Server processes.

• Do not place other files in these directories.

Use Best Practices for Working with SAS Folders

The folders that appear on the **Folders** tab of SAS Management Console are used by client applications to store BI content and system information. Inappropriate renaming, deleting, or moving of these folders or their contents could cause client applications to malfunction.

When interacting with folders in SAS Management Console, be sure to follow the best practices that are provided in Chapter 18, "Working with SAS Folders". If you need to move or copy the contents of these folders, use the procedures that are provided in Chapter 20, "Promotion Tools Overview".

Use Usermods Files When Customizing Autoexec Files, Server Configuration Files, and Start-up Scripts

If you need to customize a configuration file, autoexec file, or start-up script for a SAS server, do not directly modify the file or script. Instead, add your customizations to the corresponding file that is named *server-name*_usermods.cfg, autoexec_usermods.sas, *server-name*_usermods.bat, or *server-name*_usermods.sh.

Use of these files prevents your customizations from being overwritten when a new SAS release is installed.

Be Aware of the SAS Virtualization Environment Policy

Do not make changes that affect the responsiveness of virtual machines that are running SAS servers. For more information, see SAS Product Support for Virtualization Environments.

Part 2

Starting, Stopping, and Checking the Status of Servers

Chapter 6

Operating Your Servers

Overview of Server Operation

About Required Servers

In order for clients to access the SAS Intelligence Platform, the following components must be running on network-accessible machines:

- a SAS Metadata Server

- the SAS object spawner, which acts as a listener for SAS Workspace Servers, SAS Pooled Workspace Servers, and SAS Stored Process Servers

- a SAS Web Infrastructure Platform Data Server

- middle-tier servers including the SAS Web Application Server, SAS Web Server, SAS Environment Manager server, and the supporting JMS Broker and Cache Locator components

- an instance of SAS Environment Manager Agent on each server-tier and middle-tier machine

You might also have the following components on network-accessible machines:

- a SAS OLAP Server, a SAS/SHARE server, a SAS/CONNECT spawner, and SAS Distributed In-Process Scheduler Job Runner

- a SAS Deployment Tester server, which is used to run the SAS Deployment Tester utility

- the SAS Deployment Agent, which is required for some administration activities such as creating new servers, configuring middle-tier clustering, and using the Deployment Backup and Recovery tool on page 387.

Starting Servers in the Correct Order

Because of dependencies, it is important to start the servers in the correct order. The following table specifies server dependencies and a recommended start-up order:

Table 6.1   Server Dependencies and Recommended Start-Up Order

Start Order	Server or Service	Tier	Dependencies
1	SAS Metadata Server***	Server tier	
2	SAS Web Infrastructure Platform Data Server	Server tier	
3	SAS OLAP Server	Server tier	SAS Metadata Server
4	SAS object spawner	Server tier	SAS Metadata Server
5	SAS/SHARE server	Server tier	SAS Metadata Server
6	SAS/CONNECT spawner	Server tier	SAS Metadata Server

Start Order	Server or Service	Tier	Dependencies
7	SAS Deployment Tester server	Server tier	SAS Metadata Server
8	SAS Distributed In-Process Scheduler Job Runner	Server tier	SAS Metadata Server
9	JMS Broker††	Middle tier	
10	Cache Locator ††	Middle tier	
11	SAS Web Server††***	Middle tier	
12	SAS Web Application Server	Middle tier	Cache Locator
13	SAS Environment Manager server	Middle tier	SAS Web Infrastructure Platform Data Server † SAS Web Application Server†
14	SAS Environment Manager Agent	Server and middle tier	
15	SAS Deployment Agent	Server and middle tier	

  * In clustered configurations, make sure that all metadata server nodes are running before you start dependent components.

 ** In SAS 9.4M3, the documentation was changed to recommend starting the SAS Metadata Server first, followed by the SAS Web Infrastructure Platform Data Server (instead of the reverse). This order is easier and more logical for deployments in which the metadata server is installed on a separate machine from the other server-tier components.

*** In SAS 9.4M1, the documentation was changed to recommend that the SAS Web Server be started before the SAS Web Application Server. This start-up order helps ensure optimum performance when web applications are initialized. The sas.servers script has also been changed to incorporate the new order.

  † The SAS Environment Manager server can start without these components. However, the SAS Environment Manager application requires them in order to provide full functionality.

†† In clustered middle tier environments, start the JMS Broker, Cache Locator, and web server on each middle tier node before moving to the next middle tier node.

Servers should be stopped in the inverse order, with the SAS Web Infrastructure Platform Data Server and metadata server stopped last.

About Server Start-Up Parameters

The start-up parameters for SAS servers are stored in configuration files. For a list of these configuration files and their locations, see "Reference: Configuration Files for SAS Servers" on page 447. For detailed information about start-up parameters for each server, see the following documentation references:

- SAS Metadata Server, SAS OLAP Server, and SAS object spawner:

  - Chapter 32, "Options for the Metadata Server Invocation Command," on page 469

  - "Overview of System Options for SAS Application Server Components" in *SAS Intelligence Platform: Application Server Administration Guide*

  - "Spawner Invocation Options" in *SAS Intelligence Platform: Application Server Administration Guide*

- SAS/CONNECT spawner and SAS/SHARE Server: *SAS/CONNECT User's Guide* and *SAS/SHARE User's Guide*.

- SAS Web Application Server, SAS Web Server, JMS broker, and Cache Locator: *SAS Intelligence Platform: Middle-Tier Administration Guide*

- SAS Environment Manager server and SAS Environment Manager Agent: SAS Environment Manager Help

Methods for Operating Servers

The following table specifies the operation methods that are available for each server:

Table 6.2   Available Methods of Operation for Servers

Start Order	Server or Service	Tier	Methods of Operation				
			Windows Service	**sas.servers Script (for UNIX and z/OS)**	**Individual Server Script**	**SAS Environment Manager**	**SAS Management Console**
1	SAS Metadata Server	Server tier	✓	✓	✓	✓	✓ (except start-up)
2	SAS Web Infrastructure Platform Data Server	Server tier	✓	✓	✓		
3	SAS OLAP Server	Server tier	✓	✓	✓	✓	✓ (except start-up)
4	SAS object spawner	Server tier	✓	✓	✓	✓	✓ (except start-up)
5	SAS/SHARE server	Server tier	✓	✓	✓		
6	SAS/CONNECT spawner	Server tier	✓	✓	✓	✓	✓ (except start-up)
7	SAS Deployment Tester server	Server tier	✓		✓		
8	SAS Distributed In-Process Scheduler Job Runner	Server tier	✓	✓	✓		

Start Order	Server or Service	Tier	Methods of Operation				
			Windows Service	sas.servers Script (for UNIX and z/OS)	Individual Server Script	SAS Environment Manager	SAS Management Console
9	JMS Broker	Middle tier	✓	✓	✓ (UNIX only)		
10	Cache Locator	Middle tier	✓	✓	✓ (UNIX only)		
11	SAS Web Server	Middle tier	✓	✓	✓	✓	
12	SAS Web Application Server	Middle tier	✓	✓	✓	✓	
13	SAS Environment Manager server	Middle tier	✓	✓			
14	SAS Environment Manager Agent	Server tier and middle tier	✓	✓		✓	
15*	SAS Deployment Agent	Server tier and middle tier	✓			✓	

* You can also use SAS Deployment Manager to start and stop the SAS Deployment Agent. See "Starting and Stopping the SAS Deployment Agent" on page 83.

> *Note:* On z/OS systems, SAS recommends that you operate servers as started tasks. See "Starting and Stopping SAS Servers on z/OS" on page 80.

For details about each operation method, see the following topics:

- "Running Servers as Windows Services" on page 66
- "Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers" on page 67
- "Using Scripts to Operate SAS Servers Individually" on page 70
- "Using SAS Environment Manager to Operate Servers" on page 75
- "Using SAS Management Console to Operate SAS Servers" on page 76

See Also

- "Running Servers as UNIX Daemons" on page 81

Running Servers as Windows Services

On Windows systems, the SAS Deployment Wizard installs Windows services for all of the servers and services that are listed in Table 6.1 on page 62. These services start automatically when you restart the machines. See "Check Prerequisite Servers" in *SAS Intelligence Platform: Middle-Tier Administration Guide* for more information.

The service names have the following format:

SAS [deployment-name-and-level**]** <server-context -> server-name

For example, your services might be displayed in Windows Services Manager as follows:

```
SAS [Config-Lev1] Object Spawner
SAS [Config-Lev1] SASApp - OLAP Server
SAS [Config-Lev1] SASMeta - Metadata Server
```

You can use the following methods to operate Windows Services:

• In Windows Services Manager, right-click the service and select an operation.

• Use the appropriate batch script that is provided by SAS to manage the service from the command line.

• Use the Windows Net command, which has the following syntax:

net start│stop│pause│continue "service-name**"**

For example, if the service name for the object spawner is **SAS [Config-Lev1] Object Spawner**, you would use the following commands to start and stop the service:

• **net start "SAS [Config-Lev1] Object Spawner"**

• **net stop "SAS [Config-Lev1] Object Spawner"**

The following table lists the actions that are available for each service.

Table 6.3   *Available Windows Service Actions*

Server	Available Net Commands
SAS Deployment Agent, SAS Metadata Server, SAS OLAP Server, and SAS object spawner	Start, stop, pause, resume (continue), and restart
SAS/SHARE server, SAS/CONNECT spawner, SAS Deployment Tester server, SAS Distributed In-Process Scheduler Job Runner, JMS Broker, Cache Locator, SAS Web Application Server, SAS Web Server, SAS Environment Manager server, and SAS Environment Manager Agent	Start, stop, and restart

Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers

About the sas.servers Script

During installation, the SAS Deployment Wizard creates a sas.servers script on each UNIX and z/OS server-tier and middle-tier machine. With this script, you can start, stop, or restart all of the essential SAS servers and spawners on the machine in the correct order by using a single command. You can also display the status of all of the servers on a machine by using a single command.

Some servers are started directly by the sas.servers script. Other servers are started by the sas.servers.pre and sas.servers.mid scripts, which are called by sas.servers. The following table shows the script names, the components that are included in each script, and the order in which the components are started.

Script	Tier	Start-Up Order
sas.servers.pre (called by sas.servers)	Server tier	SAS Web Infrastructure Platform Data Server [*]
sas.servers	Server tier	SAS Metadata Server[**], SAS OLAP Server, SAS object spawner, SAS/SHARE server, SAS/CONNECT spawner, and SAS Distributed In-Process Scheduler Job Runner
sas.servers.mid (called by sas.servers)	Middle tier	JMS Broker, Cache Locator, SAS Web Server[***], SAS Web Application Server, and SAS Environment Manager server
sas.servers.mid (called by sas.servers)	Server and middle tier	SAS Environment Manager Agent

[*] The SAS Deployment Tester can be added to the sas.servers script on server-tier machines. See "Regenerate a sas.servers Script" on page 82.

[**] In clustered metadata server configurations, make sure that all metadata server nodes are running before you run sas.servers on the server tier.

[***] Effective with SAS 9.4M1, the sas.servers.mid script starts the SAS Web Server before the SAS Web Application Server. This start-up order helps to ensure optimum performance when web applications are initialized.

Note:

- If needed, you can use the sas.servers.pre or sas.servers.mid script to start a subset of servers. However, make sure that you follow the start-up order that is shown in the preceding table.

- Other servers might also be included in the scripts, depending on what SAS applications you have configured.

- The sas.servers script does not include SAS Deployment Agent. In order to include SAS Deployment Agent in the sas.servers script, refer to the 'Adding

Services to the SAS Deployment Agent (UNIX only)' section of the *SAS Deployment Wizard and SAS Deployment Manager 9.4: User's Guide*, available at http://support.sas.com/documentation/installcenter/en/ikdeploywizug/66034/PDF/default/user.pdf.

- To use the sas.servers script on z/OS systems, you must run the servers under UNIX System Services (USS). On z/OS systems, SAS recommends operating servers as started tasks. See "Starting and Stopping SAS Servers on z/OS" on page 80.

CAUTION:

You should not directly update the sas.servers script. If the script needs to be updated (for example, to add new servers or remove servers), then regenerate the script by using generate_boot_scripts.sh. For details, see "Regenerating a sas.servers Script" on page 82.

Use the sas.servers Script to Start, Stop, or Restart All Servers or to Display the Status of Servers

To use the sas.servers script to start, stop, restart, or display the status of all of essential servers on a UNIX or z/OS server-tier or middle-tier machine, follow these steps:

1. Log on to the host machine as the SAS Installer user (for example, sas).

2. Go to the top level of your configuration directory (for example, *SAS-configuration-directory*/**Lev1/**).

3. Type **./sas.servers** followed by one of the following arguments:

start
starts the servers and spawners in the appropriate order, as shown in Table 6.1 on page 62.

Note: One exception is that if the metadata server and SAS Web Infrastructure Platform Data Server are installed on the same machine, the sas.servers script starts the SAS Web Infrastructure Platform Data Server first.

Note: In multiple-machine deployments, make sure that the metadata server is running before you execute the script on server-tier machines that contain dependent components (including objects spawners, OLAP servers, SAS/CONNECT spawners, and SAS/SHARE servers). In clustered configurations, all of the metadata server nodes must be running before you start these components.

stop
stops the servers and spawners in the inverse order that is shown in Table 6.1 on page 62.

Note: When you stop the metadata server, any dependent servers and spawners (whether they are on the same host machine or a different machine) stop running.

Note: Prior to the February 15, 2022 release of SAS 9.4M7, the sas.servers script manages SAS Web Application Server instances with the Pivotal tc server. Thereafter, the sas.servers script manages SAS Web Application Server instances with Apache Tomcat. In releases of SAS 9.4M7 where SAS Web Application Server instances are managed by Apache Tomcat, the following error now results when the sas.servers script to stop the servers stops an instance of a SAS Web Application Server:

```
ERROR [Catalina] No shutdown port configured. Shut down server
through OS signal. Server not shut down. The stop command failed.
Attempting to signal the process to stop through OS signal. Tomcat
did not stop in time. PID file was not removed. To aid diagnostics a
thread dump has been written to standard out.
```

Contrary to what this error message states, the SAS Web Application Server instance has been stopped. Also, no shutdown port is configured by design so this is not a problem. See for information about eliminating or reducing the chance of receiving this error message.

restart
  stops and then starts the servers and spawners in the appropriate order.

status
  displays the current status of the servers and spawners in the following format:

  server-name server-instance (*process-ID*) **is running│is stopped**.

  For example, the following messages mean that two instances of the SAS OLAP Server are configured on the machine. The first instance is running, and the second instance is stopped:

```
SAS OLAP Server 1 (pid 29529) is running
SAS OLAP Server 2 (pid 29530) is stopped
```

Note: To use the script on z/OS systems, you must run the servers under UNIX System Services (USS). On z/OS systems, SAS recommends operating servers as started tasks.

Install sas.servers as a Boot Script

On UNIX machines, you can choose to install the sas.servers script as a boot script so that the servers start automatically when you start the machine. The comments in the sas.servers script contain installation instructions that are specific to your operating environment.

Troubleshooting the sas.servers Script

If you are running the February 15, 2022 release or later of SAS 9.4M7 and you receive an error message when running the sas.servers script to stop all of the SAS servers, then you might want to extend the time-out value for the SAS Web Application Server instances. This error message results from the change to use Apache Tomcat to manage SAS Web Application Server instances, beginning with the Februrary 15, 2022 release of SAS 9.4M7.

Here is the error message: **ERROR [Catalina] No shutdown port configured. Shut down server through OS signal. Server not shut down. The stop command failed. Attempting to signal the process to stop through OS signal. Tomcat did not stop in time. PID file was not removed. To aid diagnostics a thread dump has been written to standard out.**

Note: Contrary to what this error message states, the SAS Web Application Server instance has been stopped. Also, no shutdown port is configured by design so this is not a problem.

To eliminate or reduce the chance of receiving this error message, extend the time-out value for each instance of the SAS Web Application Server. To extend the time-out value, use the catalina.sh script rather than the appsrvconfig.sh script to stop each instance. You cannot use the sas.servers script to stop all of the servers in this scenario. You must stop each server individually in order to change the time-out value for the SAS Web Application Server instances. Here are the steps:

1. See "Use Individual Scripts to Operate Servers" on page 70 for instructions for stopping each SAS server. Stop each SAS server as noted in the instructions except for the SAS Web Application Server.

2. To stop the SAS Web Application Server:

   a. Navigate to the location of the catalina.sh script: **SAS-configuration-directory/Lev1/Web/WebAppServer/WebAppServer_name/bin/**.

   b. Run the following command: **catalina.sh stop timeout_value_seconds**

      For example, here is the command to extend the time-out value to 1 hour: **catalina.sh stop 3600**

Using Scripts to Operate SAS Servers Individually

About Using Scripts to Operate Servers

The configuration directory for each SAS server and spawner contains a script that enables you to operate the server or spawner individually from a command line.

On Windows machines, depending on the options that were selected during installation, the SAS Deployment Wizard might have also added shortcuts to your **Start** menu to enable you to start, stop, pause and resume servers and spawners.

On UNIX and z/OS systems, you can use the sas.servers script to start and stop servers with a single command. For more information, see "Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers" on page 67. On z/OS systems, SAS recommends operating servers as started tasks. See "Starting and Stopping SAS Servers on z/OS" on page 80.

For information about the start-up parameters for each server, see the documentation references that are provided in "Methods for Operating Servers" on page 64.

Note: If you need to customize a server script, do not directly update the script. Instead, add your customizations to the corresponding file that is named **server-name**_usermods_.bat or **server-name**_usermods.sh. Use of these files prevents your customizations from being overwritten when a new SAS release is installed.

Use Individual Scripts to Operate Servers

You can use scripts to start, stop, and restart servers individually from the DOS command line, the UNIX console, or the z/OS console. The scripts also enable you to install or remove a server or spawner as a Windows service. In some cases you can also pause and resume the server and display the server's status.

Follow these steps:

1. On Windows, log on as a user who is a member of the host operating system's Administrators group. On UNIX or z/OS, log on as the SAS Installer user (for example, sas).

2. Use one of the following scripts, which are located in the path *SAS-configuration-directory*/Lev1:

Table 6.4   *Scripts for Operating Servers and Spawners Individually*

Server or Service	Details
SAS Metadata Server	Script location: *SAS-configuration-directory*/Lev1/SASMeta/MetadataServer/
	Windows command: **MetadataServer.bat start|stop|restart|pause|resume|status|backup|optimizeIMDB|recover|startNoCluster|stopCluster|install|remove**
	UNIX command: **MetadataServer.sh start|stop|restart|pause|resume|status|backup|optimizeIMDB|recover|startNoCluster|stopCluster**
SAS Web Infrastructure Platform Data Server	Script location: *SAS-configuration-directory*/Lev1/WebInfrastructurePlatformDataServer/
	Windows command: **webinfdsvrc.bat start|stop|restart|pause|resume|status|install|remove**
	UNIX command: **webinfdsvrc.sh start|stop|restart|pause|resume|status**
SAS OLAP Server	Script location: *SAS-configuration-directory*/Lev1/SASApp/OLAPServer/
	Windows command: **OLAPServer.bat start|stop|restart|pause|resume|status|install|remove**
	UNIX command: **OLAPServer.sh start|stop|restart|pause|resume|status**
SAS object spawner	Script location: *SAS-configuration-directory*/Lev1/ObjectSpawner/
	Windows command: **ObjectSpawner.bat start|stop|restart|pause|resume|status|install|remove**
	UNIX command: **ObjectSpawner.sh start|stop|restart|pause|resume|status**
SAS/SHARE server	Script location: *SAS-configuration-directory*/Lev1/ShareServer/
	Windows command: **ShareServer.bat start|stop|restart|pause|resume|status|install|remove**
	UNIX command: **ShareServer.sh start|stop|restart|pause|resume|status**
SAS/CONNECT spawner	Script location: *SAS-configuration-directory*/Lev1/ConnectSpawner/
	Windows command: **ConnectSpawner.bat start|stop|restart|pause|resume|status|install|remove**
	UNIX command: **ConnectSpawner.sh start|stop|restart|pause|resume|status**

Server or Service	Details
SAS Deployment Tester server	Script location: **SAS-configuration-directory/Lev1/ DeploymentTesterServer/** Windows command: **DeploymentTesterServer.bat start|stop|restart| pause|resume|status|install|remove** UNIX command: **DeploymentTesterServer.sh start|stop|restart| pause|resume|status**
SAS Distributed In-Process Scheduler Job Runner	Script location (UNIX only): **SAS-configuration-directory/Lev1/Web/ Applications/SASWIPSchedulingServices9.4/dip** Windows command: (Use Windows service) UNIX command: **DIPJobRunner.sh start|stop**
JMS Broker	When you use **appsrvconfig.cmd** or **appsrvconfig.sh** to start, stop, or restart the SAS Web Application Server, the JMS Broker and Cache Locator are also started, stopped, or restarted. Use the following information if you need to start or stop the JMS Broker individually: • Script location (UNIX only): **SAS-configuration-directory/Lev1/Web/ activemq/bin/** • Windows command: (Use Windows service) • UNIX command: **activemq start|stop|status|restart**

Cache Locator	When you use **appsrvconfig.cmd** or **appsrvconfig.sh** to start, stop, or restart the SAS Web Application Server, the JMS Broker and Cache Locator are also started, stopped, or restarted.
	
	Use the following information if you need to start or stop the Cache Locator individually:
	
	• Script location (UNIX only): **SAS-configuration-directory/Lev1/Web/ geode/instances/ins_port/**
	
	   Note: For releases prior to SAS 9.4M8, the script location is **SAS-configuration- directory/Lev1/Web/gemfire/instances/ins_port/**.
	
	• Windows command: (Use Windows service)
	
	• UNIX command: **geode-locator.sh start|stop|status|restart**
	
	   Note: For releases prior to SAS 9.4M8, the UNIX command is **gemfire- locator.sh start|stop|status|restart**.

SAS Web Server	Script location: **SAS-configuration-directory/Lev1/Web/WebServer/bin/** Windows command: **powershell httpdctl.ps1 start|stop|restart** *Note:* The httpdctl.ps1 is a Windows PowerShell script. You might need to set the Powershell execution policy to remotesigned. See "Use the httpdctl Command" in *SAS Intelligence Platform: Middle-Tier Administration Guide* for more information. UNIX command: **httpdctl start|stop|restart**

Server or Service	Details
SAS Web Application Server	Script location: *SAS-configuration-directory*/Lev1/Web/Scripts/AppServer/
	Windows command: **appsrvconfig.cmd start|stop|restart**
	UNIX command: **appsrvconfig.sh start|stop|restart**
	These scripts start, stop, or restart all servers in the cluster. They also start, stop, or restart the JMS Broker and the Cache Locator.
	Note: To start or stop an individual instance of SAS Web Application Server, use SAS Environment Manager.

Script Option Descriptions

The following list describes the options for the server scripts. The table in the preceding topic specifies which options are available for each server.

Note: The hyphen in front of each argument is optional.

-start

starts the server or spawner.

On clustered metadata servers, use this command on the host machine for each node. The nodes can be started in any order. Make sure all the metadata server nodes are running before you start dependent components.

-stop

stops the server or spawner.

For clustered metadata servers, use this command if you want to stop only an individual node in the cluster. To stop the entire cluster, use **-stopCluster**.

Note: Stopping a non-clustered metadata server (or stopping all the nodes in a clustered metadata server) stops other components, including object spawners, OLAP servers, SAS/CONNECT spawners, and SAS/SHARE servers.

-restart

stops and then starts a running server or spawner.

-pause

causes the server or spawner to finish processing all active client sessions and to stop accepting new requests.

When this option is used with the metadata server, the server is placed in the Offline state.

To pause a clustered metadata server, use this command on any node in the cluster. All of the nodes in the cluster are paused automatically.

-resume

resumes the server or spawner process when it is in the paused state.

When this option is used with the metadata server, the server is returned to the Online state if the current state is Offline or Administration.

To resume a clustered metadata server, use this command on any node in the cluster. All of the nodes in the cluster are resumed automatically.

-status
>> displays the current status of the server or spawner.

-install
>> installs the server or spawner as a Windows service.

-remove
>> removes the server or spawner service from Windows.

Additional Script Options for the SAS Metadata Server

The metadata server script has the following additional options:

-backup
>> uses the metadata server's backup facility to copy all of your metadata repositories, the repository manager, and the contents of the metadata server's configuration directory to the configured backup location. The backup process executes in a separate thread while the metadata server is running. Therefore, the server does not need to be paused. For more information, see Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169.
>>
>> *CAUTION:*
>>> **For production backups, it is highly recommended that you follow the best practices that are provided in this guide.** See Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153.

-optimizeIMDB
>> optimizes the usage of memory on the metadata server. This option analyzes the handling of character variables, checking for items such as unnecessary indexes, long character variables that can be stored as variable-length strings, and duplicate strings. To apply the results of the analysis, you must pause and resume (or restart) the server after executing this command option.
>>
>> You cannot use this option on a server that has been started with clustering. However, you can use it on a single node that has been started without clustering.
>>
>> For more details, see "Using optimizeIMDB to Optimize and Report on Metadata Server Memory Usage" on page 208.

-recover
>> starts a server that is not currently running, and then restores the server's metadata repository from the most recent backup. You can use this option either on an unclustered server or on a single node within a cluster. This option provides an easy way to recover a server or a node that is unresponsive. After a node is started and its repository has been restored, the node rejoins the cluster and its repository is updated to match the other nodes in the cluster.
>>
>> *Note:* If the server or node is already running, the –recover request is ignored.
>>
>> This option is new with SAS 9.4M1.
>>
>> For important information about best practices in recovery situations, see Chapter 13, "What to Do If the SAS Metadata Server Is Unresponsive," on page 185. For additional details about clustered servers, see "Recovering a Clustered Metadata Server" on page 223.

-startNoCluster
>> starts a single node in a metadata server cluster. The node starts as a single, non-clustered metadata server that is paused to the Administration state. This action is useful when you want to perform one of the following administrative tasks on a node:

- perform a metadata server recovery using SAS Management Console

- back up the metadata server with the REORG option

- run the optimizeIMDB command option of the metadata server script

- run the Metadata Analyze and Repair tools

After you perform one of these functions, you must restart (or stop and start) the node to place it in cluster mode as the controller node. Then start the other nodes in the cluster. The controller node updates the other nodes with the new data from the recovery, REORG, optimizeIMDB, or analyze and repair operation.

-stopCluster
stops the entire metadata server cluster. Use this command on any one of the nodes in the cluster, and all of the nodes in the cluster stop automatically.

See Also

Using SAS Environment Manager to Operate Servers

The SAS Environment Manager application can be used to operate the following servers on the SAS server tier and middle tier:

- SAS Metadata Server

- SAS OLAP Server

- SAS object spawner

- SAS/CONNECT spawner

- SAS Web Application Server (which appears in SAS Environment Manager as SpringSource tc Runtime)

- SAS Web Server (which appears in SAS Environment Manager as vFabric Web Server)

- SAS Deployment Agent

To use SAS Environment Manager to start, stop, restart, pause, or resume a server, follow these steps:

1. Log on to SAS Environment Manager.

2. Click the **Resources** tab.

3. In the list of servers, click the server that you want to operate.

4. On the detailed page for the server, click **Control**.

5. In the **Control Action** drop-down box, select the action that you want to take (for example, **Start**, **Stop**, **Restart**, **Pause**, or **Resume**).

6. Click ▶ to the right of the drop-down box.

See Also

SAS Environment Manager User's Guide

Using SAS Management Console to Operate SAS Servers

About Using SAS Management Console to Operate SAS Servers

SAS Management Console can be used to operate and monitor most SAS servers on the SAS server tier. For a list of these servers, see Table 6.2 on page 64..

Obtain the Appropriate User Credentials, and Clear Cached Credentials (If Necessary)

To connect to a server or spawner in SAS Management Console, you must have the appropriate user credentials. Membership in the SAS Administrators group typically provides sufficient rights and permissions. For details, see "Who Can Do What in the Server Manager Plug-in" on page 41.

If you attempt to connect to a server or spawner and you are denied access, make sure you have the appropriate credentials. After your credentials are updated, be sure to clear the credentials cache before you try to connect again. To clear the credentials cache, click **File** ⇨ **Clear Credentials Cache** in SAS Management Console.

Use SAS Management Console to Operate the SAS Metadata Server

You can use SAS Management Console to stop, pause, or resume the SAS Metadata Server. You can also use SAS Management Console to reset the metadata server in order to recover memory on the host machine.

Note: Before using these features to operate the metadata server, see "About SAS Metadata Server States" on page 86.

Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. On the **Plug-ins** tab, expand the **Metadata Manager** node, and right-click the **Active Server** node.

3. Select one of the following actions:

   Stop
   completely stops the metadata server.

   Note:

   - Stopping the metadata server stops other components, including object spawners, OLAP servers, SAS/CONNECT spawners, and SAS/SHARE servers.

   - To start the metadata server after stopping it, you must use a script or Windows services. If you want to be able to resume processing from SAS Management Console, use **Pause** ⇨ **Offline** instead.

   Pause ⇨ **Administration**
   pauses the metadata server to the Administration state. In the Pause Comment window, you can enter a comment that provides information about the pause action. When a user who is not in the **Metadata Server: Unrestricted** role

attempts to read or write metadata, the comment is appended to the appropriate message and displayed as follows: **Error accessing metadata: The GetMetadataObjects request failed because the server is paused for ADMINISTRATION use. The reason for the server pause is** *comment*.

Pause ⇨ Offline
pauses the metadata server to the Offline state. In the Pause Comment window, you can enter a comment that provides information about the pause action. When a client attempts to read or write metadata, the comment is appended to the appropriate message and displayed as follows:

Error accessing metadata: The GetMetadataObjects request failed because the server is paused for OFFLINE use. The reason for the server pause is *comment*.

Pause ⇨ Read Only
pauses the metadata server to the Read-Only state. In the Pause Comment window, you can enter a comment that provides information about the pause action.

Resume
returns the metadata server to the Online state (if the current state is Offline, Administration, or Read Only). The metadata server resumes accepting client requests to read and write metadata and re-opens repository containers as clients request metadata from them.

Reset
unloads from memory and closes the containers of all registered repositories in order to recover memory on the metadata server host. The metadata server then re-opens repository containers as clients request metadata from them. This option is useful if the host computer experiences an out-of-memory condition.

See Also

Use SAS Management Console to Operate the SAS Object Spawner or the SAS/CONNECT Spawner

You can use SAS Management Console to stop, pause, resume, quiesce, or refresh the object spawner or the SAS/CONNECT spawner. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. On the **Plug-ins** tab, expand the **Server Manager** node.

3. Take one of the following actions:

   - Expand the **Object Spawner** node. Then right-click the object spawner instance that you want to administer, and click **Connect**.

   - Right-click the **Connect Spawner** node, and click **Connect**.

4. After you are connected, you can select any of the following actions:

   Refresh
   updates the SAS Management Console display with current information about the spawner.

   Disconnect
   discontinues your current connection to the spawner.

Stop
stops the spawner process, including any active client requests.

Note: If you want active requests to be completed first, use **Quiesce**.

Pause
causes the spawner to finish processing all active client requests and to stop accepting new requests.

Resume
resumes the spawner process when it is in the paused state.

Quiesce
stops the spawner process after the spawner has finished processing all active client requests.

Refresh Spawner
re-initializes the spawner to pick up and apply any changes that have been made to the server or spawner configurations. As part of the refresh process, the spawner quiesces any servers that it has started. These servers shut down when their clients have completed their work.

Note: If the spawner is in a paused state when the **Refresh Spawner** action occurs, the spawner remains paused after it is re-initialized.

The following additional option (new with SAS 9.4) is available for the object spawner:

Refresh Application Server
is similar to the Refresh Spawner operation, except that the refresh affects only a single SAS Application Server context that you select. After you select a server context (for example, SASApp), the spawner picks up and applies any changes that have been made to server configurations in that context. As part of the refresh process, the spawner quiesces any servers in that context that it has started. These servers shut down when their clients have completed their work.

Note: If the spawner is in a paused state when the **Refresh Application Server** action occurs, the spawner remains paused after it is re-initialized.

Clear Spawner Credentials Cache (or **Clear Server Credentials Cache**)
clears the location in which the spawner's credentials decisions are stored. If the spawner denies access to a server because of the lack of ReadMetadata permissions, and the user is then granted the needed permission, you must select this option to clear the cache before the user tries to connect again.

Use SAS Management Console to Stop, Pause, Resume, Quiesce, or Refresh the SAS OLAP Server

You can use SAS Management Console to stop, pause, resume, quiesce, or refresh the SAS OLAP Server. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. On the **Plug-ins** tab, expand the **Server Manager** node, the application server node (for example, **SASApp**), and the node for the logical OLAP server.

3. Right-click the OLAP server instance that you want to administer, and click **Connect**.

4. After you are connected, you can select any of the following actions:

Refresh
updates the SAS Management Console display with current information about the OLAP server.

Disconnect
discontinues your current connection to the OLAP server.

Stop
stops the OLAP server process, including any active client requests.

Note: If you want active sessions to be completed first, use **Quiesce**.

Pause
causes the OLAP server to finish processing all active sessions and to stop accepting new client requests.

Resume
resumes the OLAP server process when it is in the paused state.

Quiesce
stops the OLAP server process after the server has finished processing all active sessions.

Note: To start the OLAP server after quiescing it, you must use a script or Windows services. If you want to be able resume processing from SAS Management Console, use **Pause** instead.

Before you stop an OLAP server, you should first stop the active sessions, as follows:

1. In SAS Management Console, expand the **Monitoring** node and the **SAS OLAP Server Monitor** node.

2. Right-click the name of the logical SAS OLAP Server, and select **Connect**.

3. In the window that appears, enter your user name and password and click **OK**.

4. Right-click the name of each session that you want to close, and select **Close Session**.

Use SAS Management Console to Control Processes on Servers

You can use SAS Management Console to stop, pause, resume, or quiesce a process on a workspace server, pooled workspace server, or stored process server. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. On the **Plug-ins** tab, expand the **Server Manager** node and the application server node (for example, **SASApp**).

3. Expand the node for the logical server that is running the process, and then expand the node for the server instance that is running the process. The server's available connections are displayed in the server tree and in the right pane.

4. Highlight the appropriate host machine name, as shown in this example.

Right-click the mouse, and click **Connect**. For each process that is active for that connection, the right pane displays the process ID, the process owner, and the port, as shown in this example:

5. In the Server Manager tree, expand the node for the connection that you are working with. The active processes for the connection are displayed in the tree.

6. Highlight the process that you want to stop, pause, resume, or quiesce, as shown in this example:

In the right pane, the following tabs become active: **Clients**, **Options**, and **Loggers**. For pooled workspace servers and stored process servers, the **Sessions** and **Log** tabs also become active.

Note: If some tabs are disabled, then you need additional user credentials. See "Who Can Do What in the Server Manager Plug-in" on page 41. For details about the tabs, see "Use the Server Manager Tabs to Obtain Monitoring Information and to Adjust Logging Levels (If Needed)" on page 115.

7. Right-click the mouse and select one of the following actions:

Stop
stops the server process, including any active client requests.

Note:

• If you want active requests to be completed first, use **Quiesce**.

Pause
causes the server process to finish processing all active client requests and to stop accepting new requests.

Resume
resumes the server process when it is in the paused state.

Quiesce
stops the server process after the server has finished processing all active client requests.

Starting and Stopping SAS Servers on z/OS

On z/OS systems, the SAS Deployment Wizard initially starts the SAS servers under UNIX System Services (USS) by using scripts that are provided for each server. After the initial installation and configuration has been completed and validated, SAS recommends that you stop the servers and restart them as started tasks.

The SAS Deployment Wizard creates started task definitions for each server. The definitions are stored in *SAS-installation-location*.W0.SRVPROC. To deploy the started tasks, see the instructions.html document that is created by the SAS Deployment Wizard.

After the started tasks are deployed, you can operate your servers as follows:

1. Log on as the SAS Installer user (for example, sas).

2. To start a server, use the following console command:

   ```
   START started-task-name
   ```

   To stop a server, use the following console command.

   ```
   STOP started-task-name
   ```

   Note:

   - You can perform the equivalent of a restart by stopping and then starting a server.

   - Stopping the metadata server stops other components, including object spawners, OLAP servers, SAS/CONNECT spawners, and SAS/SHARE servers.

3. Messages are written to the system log when the SAS Metadata Server, the SAS OLAP Server, or the SAS/SHARE server is started or stopped. Automated tools can scan the system log to determine the current operational state of these servers. The log messages are displayed with the prefix **SAH**, as shown in the following examples:

   ```
   SAH011001I SAS Metadata Server <(port-number)>, State, starting
   SAH011999I SAS Metadata Server <(port-number)>, State, running
   SAH019001I SAS Metadata Server <(port-number)>, State, stopping
   SAH019999I SAS Metadata Server <(port-number)>, State, stopped
   ```

When servers are running under USS, you can start and stop the servers as follows:

- by using individual scripts that are provided for each server. See "Using Scripts to Operate SAS Servers Individually" on page 70.

- by using the sas.servers script. See "Using the sas.servers Script on UNIX or z/OS to Start or Stop All Servers" on page 67.

These methods are not recommended for production environments.

For more information about using SAS on z/OS, see the *SAS Companion for z/OS*.

Running Servers as UNIX Daemons

On UNIX systems, as an alternative to using the sas.servers script, you can configure your system so that certain servers or spawners run as daemons. For example, to make the metadata server run as a daemon, you can copy the MetadataServer.sh script to the boot directory of your machine. Then add the needed information and links to the host's start and stop commands so that the metadata server is started at boot time and stopped at shutdown time. For more information, see your UNIX system administrator or the system administration guide for your platform.

Regenerating a sas.servers Script

About generate_boot_scripts.sh

The generate_boot_scripts.sh script enables you to automatically regenerate the sas.servers script for a particular UNIX or z/OS server-tier or middle-tier host machine. This feature is useful in the following types of situations:

- You have configured a new server or spawner on the machine. In this situation, the script generation process detects the new server or spawner and adds it to the sas.servers script.

- You have removed a server or spawner configuration from the machine. In this situation, the script generation process detects the absence of the server or spawner and removes it from the sas.servers script.

- You want to remove a server or spawner from the script so that you can operate it separately, or you want to add a server or spawner that you previously removed.

- You have changed the location of the server's log.

About the server-name.srv Files

The configuration directory on each UNIX and z/OS server-tier and middle-tier machine contains a file called *server-name*.srv. For example, the metadata server's configuration directory contains a file called MetadataServer.srv, and the workspace server's configuration directory contains a file called WorkspaceServer.srv.

The generate_boot_scripts.sh script accesses each of the *server-name*.srv files to determine which servers and spawners are to be included in the sas.servers script. If a server's *server-name*.srv file contains the keyword ENABLE, then the server is included in the script. If the file contains the keyword DISABLE, then the server is excluded from the script.

In the initial configuration, all *server-name*.srv files contain the keyword ENABLE.

The *server-name*.srv files also contain the path to the server's start-up script and the path to the server's log files.

Regenerate a sas.servers Script

To regenerate a sas.servers script, follow these steps:

1. Log on as the SAS Installer user (for example, sas).

2. If you want to remove a server or spawner from the script, or if you want to add a server or spawner that you previously removed, follow these steps:

   a. Navigate to the server's configuration directory and open the file *server-name*.srv in an editor.

      For example, if you want to remove the SAS/SHARE server from the script, you would go to **SAS-configuration-directory/Lev1/ShareServer** and open the file ShareServer.srv.

    b. Locate the line that contains the keyword ENABLE or DISABLE. If you want to exclude the server from the sas.servers script, change the keyword to DISABLE. If you want to include the server in the sas.servers script, change the keyword to ENABLE. Save the file.

3. Go to the directory ***SAS-configuration-directory*/Lev1/**.

4. Enter the following command: **./generate_boot_scripts.sh**

A new sas.servers script is created to reflect the servers that are currently configured on the machine. If any of the *server-name*.srv files contain the keyword DISABLE, the following message appears:

<number-of-disabled-servers> **SAS servers were explicitly disabled in their .srv files**

Starting and Stopping the SAS Deployment Agent

The SAS Deployment Agent is required for some administration activities such as creating new servers, configuring middle-tier clustering, and using the Deployment Backup and Recovery tool. You can use the following methods to start and stop the SAS Deployment Agent:

- Effective with SAS 9.4M3, you can use SAS Environment Manager to start and stop the SAS Deployment Agent.

- On Windows systems, the SAS Deployment Agent is typically installed as a Windows service. You can use the Windows Services Manager to start or stop the service. The service can be configured to start automatically when the machine is started.

- You can use SAS Deployment Manager to start and stop the SAS Deployment Agent as follows:

  1. Navigate to ***SAS-installation-directory*/SASDeploymentManager/ 9.4**.

  2. Launch **sasdm.exe** (Windows), **sasdm.sh** (UNIX) or **sasdm.rexx** (z/OS).

  3. Expand the **Manage SAS Deployment Agent Service** node. Then select **Start SAS Deployment Agent Service** or **Stop SAS Deployment Agent Service**.

See Also

"Manage SAS Deployment Agent Service" in the *SAS Deployment Manager: User's Guide*

Chapter 7

Checking the Status of Servers

About Checking the Status of Servers

SAS provides a number of tools that you can use to determine the status of your SAS server components. You can do the following

- use a server script to display the status of a server from the command line

- use SAS Management Console or SAS programming features to check the access state and other properties of the SAS Metadata Server and the access state of a SAS Metadata Repository

- use SAS Management Console to do the following:

- check an instance of the SAS object spawner or SAS/CONNECT spawner to ensure that you can make a connection

- submit SAS code to validate servers to ensure that they are set up correctly

- validate the SAS Content Server to determine whether it is listening on the specified host and port and that the first configured base path exists on the server

To run more extensive validation tests on the SAS Content Server, you can use the SAS Deployment Tester plug-in in SAS Management Console or the SAS Deployment Tester batch clients. See or the SAS Management Console Help. You can also use SAS programming features to check the status of the metadata server, metadata repositories, and the repository manager.

Note: To check the status of a SAS/SHARE server, see the *SAS/SHARE User's Guide*.

See Also

- Chapter 8, "Using the Deployment Tester," on page 97

- "Using SAS Environment Manager to Monitor SAS Servers" on page 128

Use a Server Script to Display the Status of a Server or Spawner

To display the status of a server or spawner from a command line, you can use the **-status** option of the script for that server or spawner. For details, see "Using Scripts to Operate SAS Servers Individually" on page 70.

On UNIX and z/OS, you can use the **-status** option of the sas.servers script to display the status of the servers and spawners on a machine. For details, see "Use the sas.servers Script to Start, Stop, or Restart All Servers or to Display the Status of Servers" on page 68.

Checking the Access State and Other Properties of the SAS Metadata Server

About SAS Metadata Server States

A SAS Metadata Server can exist in any of the following states:

Online
The metadata server is available to accept client requests from authenticated users.

External systems such as Windows Services Manager report the Online state as **running** or **started**.

Read Only
The metadata server remains online and continues to accept client requests, but the access state of all online metadata repositories is temporarily changed to Read Only. Client applications can read metadata. However, an error message is returned if a client application attempts to write, update, or delete metadata.

External systems such as Windows Services Manager report the Read-Only state as `paused`.

Offline

The metadata server is paused and is not available to accept client requests. When the metadata server is paused to Offline status, the server does the following:

- retains all active client connections.

- temporarily changes the access state of all repositories to Offline.

- temporarily stops receiving client requests to read or write metadata. If a client application submits a new request, the client displays a message indicating that the server is paused.

- flushes updated metadata from the server's memory and writes the updated metadata to the metadata repository.

- closes the SAS data sets that serve as repository containers.

The Offline state is useful if you need to disable users' access while you perform maintenance on the metadata server machine. External systems such as Windows Services Manager report the Offline state as `paused`.

Administration

The metadata server is paused and will accept requests only from an unrestricted administrative user. When the metadata server is paused to Administration status, the server does the following:

- retains all active client connections.

- temporarily stops receiving client requests to read or write metadata, with the following exceptions

  - users who are in the **Metadata Server: Unrestricted** role can continue to perform all functions in SAS Management Console.

  - users who are in both the **Metadata Server: Operation** and the **Management Console: Advanced** role (or who are in the SAS Administrators group) can stop, pause, resume, and reset the metadata server.

In all other situations, the client displays a message indicating that the server is paused.

The Administration state is useful when you need to disable users' access while you connect to the metadata server to perform administrative tasks such as the following:

- run diagnostic and repair tools

- upgrade metadata to a new version

- promote metadata and related content

- modify metadata for components of a SAS Application Server

In addition, the server is automatically placed in the Administration state in the following situations:

- When you execute the Metadata Analyze and Repair tools.

- If problems occur during server initialization when the server is started or resumed. When the server is in the Administration state, the administrator can log on to SAS Management Console, examine the server log to determine the source of the problem, make any necessary corrections, and resume the server.

External systems such as Windows Services Manager report the Administration state as `paused`.

Stopped
>The Stopped state terminates the SAS Metadata Server process and breaks all client connections. The metadata server, and all SAS servers that depend on it, must be started before clients can connect again.

Use SAS Management Console to Check the Access State and Other Properties of the SAS Metadata Server

To use SAS Management Console to check the access state and other properties of the SAS Metadata Server:

1. In the SAS Management Console **Plug-ins** tab, expand the **Metadata Manager** node.

2. Right-click the **Active Server** node and select **Properties**. The Active Server Properties dialog box appears.

   In this dialog box, the **General** tab displays the following:

   • the server's current state and, if the server is paused, any comments that were entered by the administrator

   • information about whether a backup is currently running

   • the server's locale, version, and alert email address

   • a button for testing alert e-mails

3. Select the **Statistics** tab to see server statistics including CPU seconds, memory utilization, thread utilization, and I/O activity.

4. Select the **Options** tab to see the properties that were specified in the metadata server configuration file (omaconfig.xml) when the server was started. If a displayed property value is blank, this means that the default value is in effect. See "Reference Information for omaconfig.xml" on page 456.

5. If you have configured a clustered metadata server, select the **Cluster** tab to see the overall status of the cluster (including the presence or absence of a quorum) and the status of each of the nodes in the cluster.

See Also
"Use SAS Management Console to Operate the SAS Metadata Server" on page 76

Use PROC METAOPERATE to Check the Access State of the SAS Metadata Server

To check the access state of the SAS Metadata Server, you can use PROC METAOPERATE to issue a STATUS action. The syntax is as follows:

```
PROC METAOPERATE
  SERVER='host-name'
  PORT=port-number
  USERID='user-ID'
  PASSWORD='password'
  PROTOCOL=BRIDGE
  ACTION=STATUS;
RUN;
```

Any user who can log on to the SAS Metadata Server can issue a STATUS action.

If the SAS Metadata Server is running normally, information similar to the following is written to the SAS log:

```
NOTE: Server dXXXX.us.company.com SAS Version is 9.4.
NOTE: Server dXXXX.us.company.com SAS Long Version is 9.04.01B0P0506201.
NOTE: Server dXXXX.us.company.com Operating System is X64_ESRV08.
NOTE: Server dXXXX.us.company.com Operating System Family is WIN.
NOTE: Server dXXXX.us.company.com Operating System Version is Service Pack 2.
NOTE: Server dXXXX.us.company.com Client is userid@DOMAIN.
NOTE: Server dXXXX.us.company.com Metadata Model is Version 15.02.
NOTE: Server dXXXX.us.company.com is RUNNING on 09May2013:10:31:26.
```

If the SAS Metadata Server is paused, information similar to the following is written to the SAS log:

```
NOTE: Server dXXXX.us.company.com SAS Version is 9.4.
NOTE: Server dXXXX.us.company.com SAS Long Version is 9.04.01B0P0506201.
NOTE: Server dXXXX.us.company.com Operating System is X64_ESRV08.
NOTE: Server dXXXX.us.company.com Operating System Family is WIN.
NOTE: Server dXXXX.us.company.com Operating System Version is Service Pack 2.
NOTE: Server dXXXX.us.company.com Client is userid@DOMAIN.
NOTE: Server dXXXX.us.company.com Metadata Model is Version 15.02.
NOTE: Server dXXXX.us.company.com is PAUSED on 09May2013:10:31:26.
```

A SAS Metadata Server that is stopped will not respond to a PROC METAOPERATE request.

For more information about PROC METAOPERATE, see *SAS Language Interfaces to Metadata*.

Checking the Access State of Metadata Repositories and the Repository Manager

About Metadata Repository Access States and Access Modes

The access state of a metadata repository indicates its current state with regard to client access. A metadata repository can exist in any of the following access states:

Online
    Clients can read, write, and update metadata.

Read Only
    Clients can read metadata but cannot write, update, or delete metadata.

Administration
    Only users who are in the **Metadata Server: Unrestricted** role can read or write metadata. For information about this role, see "Overview of Initial Roles, Groups, and Users" on page 19.

Administration (Read Only)
    Only users who are in the **Metadata Server: Unrestricted** role can read metadata. Users cannot write, update, or delete metadata.

Offline
    Clients can neither read nor write metadata to the repository.

The access state of a repository depends on both the current state of the metadata server and the repository's registered access mode. The access mode of a SAS metadata

repository is the repository's intended access state, as described in "About SAS Metadata Repository Access Modes" on page 229. When the metadata server is Online, the access state of a repository is equal to its access mode. When the metadata server is paused to the Offline, Read Only, or Administration state, the access state of a repository can change temporarily. When the metadata server is resumed, the repository's access state returns to its registered access mode.

How the Access State of the Metadata Server Affects the Access States of Metadata Repositories

Generally, the access state of a repository is the same as the access mode that is recorded in the repository's metadata. However, a repository's access state can change temporarily when the metadata server is paused to an Offline state or paused for administration. When the metadata server is resumed, the repository's access state returns to its registered access mode.

The following table shows how the access state of the metadata server affects the access state of a repository, based on the repository's access mode:

Table 7.1   *The Effect of Metadata Server States on Metadata Repository Access States*

Metadata Server Access State	Metadata Repository Registered Access Mode	Metadata Repository Access State
Online	Online	Online
Online	Administration	Administration
Online	Read Only	Read Only
Online	Offline	Offline
Administration	Online	Administration
Administration	Administration	Administration
Administration	Read Only	Administration (Read Only)
Administration	Offline	Offline
Read Only	Online	Read Only
Read Only	Administration	Administration (Read Only)
Read Only	Read Only	Read Only
Read Only	Offline	Offline
Offline	Online	Offline
Offline	Administration	Offline
Offline	Read Only	Offline

Metadata Server Access State	Metadata Repository Registered Access Mode	Metadata Repository Access State
Offline	Offline	Offline

About Repository Manager Access States

The repository manager is a library of tables that define the set of repositories that can be accessed by a SAS Metadata Server. An access state of Read Only or Offline for the repository manager is typically reflected in the following behavior:

Read Only
> Clients can read and write metadata in repositories, but they cannot create or delete repositories or change their registrations.

Offline
> The behavior can vary depending on whether a repository is available in memory. Assume that repositories are unavailable for reading or writing to all users.

Use SAS Management Console to Check the Access State of a SAS Metadata Repository

If the metadata server is online or is paused to an Administration or Read-Only state, then you can use SAS Management Console to determine the current access state of a metadata repository, as follows:

1. In the SAS Management Console **Plug-ins** tab, expand the **Metadata Manager** node.

2. To view the access state of the foundation repository and any custom repositories that you have created, select the **Active Server** node. To view the access state of project repositories, expand the **Active Server** node and select the **Project Repositories** folder.

   For each repository, the access mode and the current access state are displayed in the table view, as shown in the following example:
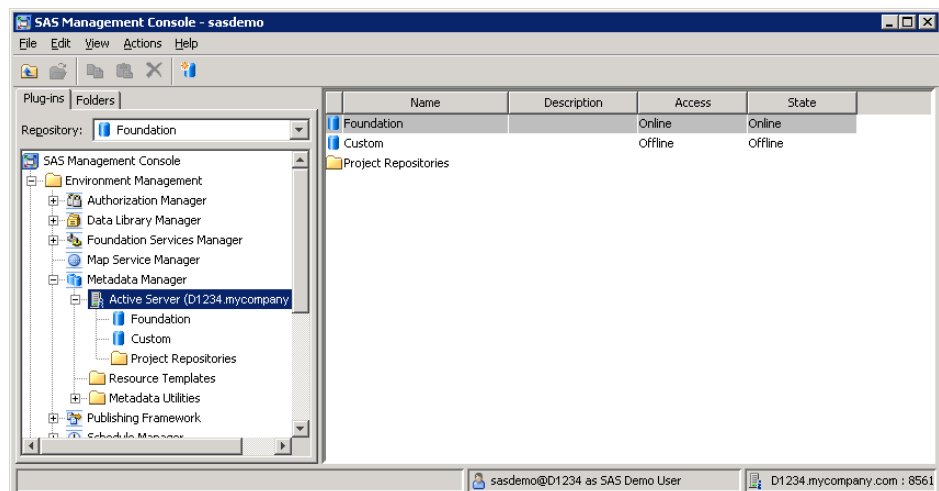
   The table lists the following information for each registered repository:

Access

indicates the access mode of the repository as it is recorded in the Access attribute of the repository's metadata definition.

State

indicates the current access state of the repository. Generally, the access state of a repository is equal to its access mode. However, if the metadata server is paused to an Offline, Administration, or Read-Only state, then the repository's access state might be changed temporarily. See "How the Access State of the Metadata Server Affects the Access States of Metadata Repositories" on page 90.

The **Access** and **State** fields are also displayed in the metadata repository properties. To see these properties, right-click the repository name in the Active Server tree, select **Properties**, and select the **Registration** tab.

Note:

- If the metadata server is paused to the Administration state, then you must be in the **Metadata Server: Unrestricted** role to view the server's status in SAS Management Console.

- If the metadata server is paused to the Offline state, then the access state of repositories cannot be displayed in SAS Management Console.

Use PROC METADATA to Check the Status of Metadata Repositories and the Repository Manager

Create and Run a PROC METADATA Program

If the metadata server is online, then you can use PROC METADATA to check the status of metadata repositories. To do so, use PROC METADATA to issue a GetRepositories method call that sets the OMI_ALL flag to **1**. The GetRepositories method is part of the SAS Open Metadata Interface.

The syntax is as follows:

```
PROC METADATA
    SERVER="host-name"
    PORT=port-number
    USERID="your-user-ID"
    PASSWORD="your-password"
    IN="<GetRepositories>
        <Repositories/>
        <Flags>1</Flags>
        <Options/>
        </GetRepositories>";
    RUN;
```

Note:

- To run this program, you must have ReadMetadata permission to all repositories on the SAS Metadata Server.

- If the metadata server is paused to an Offline state, then you cannot use PROC METADATA to issue a GetRepositories method call.

- If the metadata server is paused to an Administration state, then special user credentials are required to use PROC METADATA to issue a GetRepositories method call. The user must either (1) be a member of the **Metadata Server:**

> **Unrestricted** role or (2) be listed in adminUsers.txt, either with or without an
> asterisk.

For each repository that is registered to the metadata server, the GetRepositories method
returns the values of the following attributes:

`Id`
> specifies the repository's unique object identifier.

`Name`
> specifies the name assigned to the repository.

`Desc`
> contains a description of the repository.

`Access`
> indicates the repository's intended access mode, which is specified in the Access
> attribute of the repository's metadata definition. This field corresponds to the
> repository's **Access** field in SAS Management Console. The possible values are
> **`OMS_FULL`** (which is displayed as **`Online`** in SAS Management Console),
> **`OMS_READONLY`**, **`OMS_ADMIN`**, and **`OMS_OFFLINE`**.

`CurrentAccess`
> contains information that is used by metadata server processes. This field is
> displayed because it is sometimes used by SAS Technical Support in the resolution
> of metadata repository issues.

`PauseState`
> indicates the current access state of the repository. This field corresponds to the
> repository's **State** field in SAS Management Console. The access state depends on
> both the current state of the metadata server and the repository's registered access
> mode, as described in "How the Access State of the Metadata Server Affects the
> Access States of Metadata Repositories" on page 90. An empty string means that the
> repository is online.

`Path`
> indicates the path where the repository's data sets are located.

`RepositoryFormat`
> contains the revision level of the metadata model that is used by the repository. A
> RepositoryFormat level that is less than the level of the server can cause the
> repository to have a **`PauseState`** (access state) of Read Only.

For more information about the METADATA procedure and the GetRepositories
method, see *SAS Language Interfaces to Metadata*.

Example: PROC METADATA Output

The following is an example of output from PROC METADATA when the metadata
server is online:

```
<GetRepositories>
   <Repositories>
      <Repository Id="A00000001.A00000001"
         Name="REPOSMGR"
         Desc="The Repository Manager"
         DefaultNS="REPOS"
         RepositoryType=""
         RepositoryFormat="12"
         Access="OMS_FULL"
         CurrentAccess="OMS_FULL"
         PauseState=""
```

```
            Path="rposmgr"/>
        <Repository Id="A00000001.A50Q21KN"
            Name="Foundation"
            Desc=""
            DefaultNS="SAS"
            RepositoryType="FOUNDATION"
            RepositoryFormat="12"
            Access="OMS_FULL"
            CurrentAccess="OMS_FULL"
            PauseState=""
            Path="MetadataRepositories/Foundation"/>
        <Repository Id="A00000001.A5IPLNM1"
            Name="Test Repository"
            Desc=""
            DefaultNS="SAS"
            RepositoryType="CUSTOM"
            RepositoryFormat="12"
            Access="OMS_READONLY"
            CurrentAccess="OMS_READONLY"
            PauseState="READONLY"
            Path="MetadataRepositories/TestRepository"/>
```

This output in this example provides the following information:

- For the repository manager (**REPOSMGR**): **PauseState=""** indicates that it is available.

- For the Foundation repository: **Access="OMS_FULL"** indicates a registered access mode of Online, and **PauseState=""** indicates that the current access state is Online.

- For the Test Repository: **Access="OMS_READONLY"** indicates a registered access mode of ReadOnly, and **PauseState="READONLY"** indicates that the current access state is ReadOnly.

Check the Status of the Object Spawner or SAS/CONNECT Spawner

The SAS object spawner needs to be running in order for instances of SAS Workspace Servers, SAS Pooled Workspace Servers, and SAS Stored Process Servers to be launched, and the SAS/CONNECT spawner needs to be running in order for SAS/CONNECT server instances to be launched.

To make sure that a spawner is working, follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Server Manager Plug-in" on page 41.

2. On the **Plug-ins** tab, expand the **Server Manager** node.

3. Expand the **Object Spawner** node. Then right-click the object spawner instance that you want to check, and click **Connect**.

4. To check the status of a SAS/CONNECT spawner, right-click the spawner instance that you want to check, and click **Connect**.

Validate the SAS Metadata Server, SAS Workspace Servers, SAS Pooled Workspace Servers, SAS Stored Process Servers, SAS OLAP Servers, and SAS/CONNECT Servers

95

Validate the SAS Metadata Server, SAS Workspace Servers, SAS Pooled Workspace Servers, SAS Stored Process Servers, SAS OLAP Servers, and SAS/CONNECT Servers

To determine whether your metadata server, workspace servers, pooled workspace servers, stored process servers, OLAP servers, and connect servers are set up correctly, you can use the Validate feature. This feature submits a small SAS program to the server to determine whether it can process the code. You can validate either a logical server or a particular instance of a server.

Note: You cannot validate a workspace server that is configured to use client-side (rather than server-side) pooling. To determine whether a workspace server uses client-side pooling, right-click the server's logical server node and click **Properties**. If the properties dialog box contains a **Pooling** tab that lists at least one puddle, then the server uses client-side pooling.

To validate a server, follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Server Manager Plug-in" on page 41.

2. In SAS Management Console, expand the **Server Manager** node and the node for the application server (for example, **SASApp**).

3. To validate a logical server:

   a. Right-click the logical workspace server, logical pooled workspace server, logical stored process server, logical OLAP server, or logical connect server that you want to validate.

      Note: If the pre-assigned libraries are loaded, the credentials that you use to validate must provide access to the database and must provide the ability to load the libraries. Otherwise, an error occurs.

   b. Select **Validate**.

4. To validate a particular server instance:

   a. Expand the appropriate logical server node.

   b. In the tree that appears under the logical server definition, expand the server definition node.

   c. In the tree that appears under the server definition, select the server instance that you want to validate, as shown in this example:
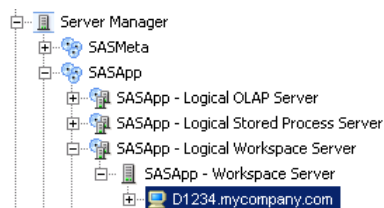
   d. Right-click the server instance and select **Validate**.

5. SAS Management Console sends a small SAS program to the server for execution. If the program executes correctly, the following message is displayed:

```
Validation Successful!
```

If the program does not execute correctly, an error message is displayed. For assistance in resolving the problem, see "Troubleshooting the SAS Server Tier" in *SAS Intelligence Platform: Installation and Configuration Guide*.

If you are denied access, make sure you have the appropriate credentials. After your credentials are updated, click **File ⇨ Clear Credentials Cache** in SAS Management Console before you try to validate again.

Note: To run more extensive validation tests, you can use the SAS Deployment Tester plug-in in SAS Management Console or the SAS Deployment Tester batch clients. These tools enable you to validate components on both the middle tier and the SAS server tier. For more information, see Chapter 8, "Using the Deployment Tester," on page 97 or the SAS Management Console Help.

Validate the SAS Content Server

If your installation includes a SAS Content Server, you can use the Validate feature to determine whether this server is set up correctly, as follows:

1. Log on to SAS Management Console, and expand the **Server Manager** node.

2. Right-click the **SAS Content Server** node and select **Validate**.

   SAS Management Console attempts to make a WebDAV connection to the server. If the connection is successful, the following message is displayed:

   ```
   Validation Successful!
   ```

   This message indicates that a WebDAV server is listening on the specified host and port, and that the first configured base path exists on the server.

   If the connection is not successful, an error message is displayed.

Note: To run more extensive validation tests on the SAS Content Server, you can use the SAS Deployment Tester plug-in in SAS Management Console or the SAS Deployment Tester batch clients. See Chapter 8, "Using the Deployment Tester," on page 97 or the SAS Management Console Help.

Chapter 8
Using the Deployment Tester

Deployment Tester Overview

What Is the Deployment Tester?

The Deployment Tester is a diagnostic tool used for assessing a SAS deployment. After an installation or upgrade, you can use the Deployment Tester to ensure that your SAS

software and critical server components have been installed and configured correctly. After making changes to your SAS deployment or to the underlying hardware or operating environment, you can use the Deployment Tester to ensure that SAS components are still in working order. The Deployment Tester is also useful for ongoing monitoring of the health of your deployment, as well as for troubleshooting when problems occur.

The Deployment Tester consists of the following components:

• the Deployment Tester server. This server is installed on each server-tier machine in a SAS deployment.

• the Deployment Tester plug-in for SAS Management Console, which enables you to run deployment tests interactively from a single interface. You can also view test results and manage test result files from the plug-in.

• the Deployment Tester batch client, which enables you to run deployment tests from a command line or on a scheduled basis. The batch client is installed on each machine where SAS Foundation is installed.

• the following test suites:

   • the SAS Installation Qualification Tool (SAS IQ) test suite, which demonstrates that your SAS software has been installed and maintained according to specifications. SAS IQ is also delivered as standalone application that can be run independently of the Deployment Tester.

   • the SAS Operational Qualification Tool (SAS OQ) test suite, which demonstrates that your Base SAS software is operational. SAS OQ is also delivered as standalone application that can be run independently of the Deployment Tester.

   • suites to test the functionality of SAS OLAP Servers, SAS Stored Process Servers, and SAS Content Servers

   • suites to check SAS licenses and their expiration dates on SAS Pooled Workspace Server and SAS Workspace Server host machines

You can use SAS OQ to create additional tests for your SAS software, and additional test suites are available for download. As tests are developed by SAS or by your organization, you can add them to the Deployment Tester.

When to Use the Deployment Tester

The validation tools that are described in Chapter 7, "Checking the Status of Servers," on page 85 are useful when you want to perform a quick check of your SAS servers. The Deployment Tester supplements the validation tools by providing more extensive testing. As a best practice, you should use the Deployment Tester at the following times to ensure that your SAS deployment is functional and operational:

• after you have used the SAS Deployment Wizard to install new software, to upgrade your SAS software, or to configure a new SAS Application Server component

• after you have made any type of change to the configuration of the SAS Content Server or a SAS Application Server component

• after you have used SAS Deployment Manager to uninstall SAS software, to remove an existing configuration, to update host name references, or to apply a hot fix

• after you have promoted server metadata from one metadata repository to another

• after you have used the SAS Migration Utility to migrate your SAS content from one SAS 9.4 deployment to another

- after you have recovered metadata repository files from a backup

Note: After you add a new server to your configuration (or install software that includes one or more additional servers), be sure to restart the Deployment Tester Server. Otherwise, the Deployment Tester will not recognize the new servers.

When your site experiences a problem with the operation of SAS software, you can use the Deployment Tester as a troubleshooting tool to help identify the cause. If necessary, you can save the test results and send them to SAS Technical Support for troubleshooting assistance.

Using the Deployment Tester on a Regularly Scheduled Basis

You might want to run the Deployment Tester on a regular schedule (for example, daily or once a week) to check the health of your system. This feature can be helpful in complying with regulatory policies. To run tests on a scheduled basis, use the batch client. See .

Prerequisites for Using the Deployment Tester

Installing JUnit

To run certain test suites, the Deployment Tester requires JUnit, which is an open-source framework for writing and running tests in the Java programming language. Before your SAS software was installed, JUnit should have been installed on each Deployment Tester server machine, as specified on the pre-installation checklist.

If JUnit is not found when you attempt to run a test that requires it, an error message appears. The message includes information about the path in which to install the JAR file for the JUnit software.

The JUnit software can be downloaded from http://support.sas.com/thirdpartysupport. On the Third-Party Software Requirements page, select the **Other Third Party Software Products** link for your SAS software release. On the SAS 9.4 Support for Additional Software Products page, note the version of JUnit that is required. Then click the **Download** link for JUnit and follow the instructions to download the software.

After downloading the software, move the JAR file to the appropriate directory. If the name of the JAR file includes a version number, you must rename the file to junit.jar.

Starting the Deployment Tester Server

The Deployment Tester server is installed on each server-tier machine in a SAS deployment. Before executing a Deployment Tester test suite, you must make sure that the Deployment Tester server is running on the host machine for the process that you are testing.

Because the Deployment Tester server is an optional server, it set up to be started manually. It is set up as a manual service on Windows machines, and it is not included in the sas.servers script on UNIX and z/OS machines. When you finish installing and configuring SAS, the server is automatically started on each server-tier and middle-tier machine so that you can test the newly installed components. After you perform these initial tests, it is recommended that you stop the server so that the resources will be available for other processes.

When you need to use the Deployment Tester plug-in or batch client, use one of the following procedures to start the Deployment Tester server:

- On Windows: Start the Deployment Tester server Windows service; start the server from the Windows **Start** menu; or go to *SAS-configuration-directory* `\Lev1\DeploymentTesterServer` and run the command `DeploymentTesterServer.bat -start`. When you are finished using the Deployment Tester, stop the server by using the same command with the `-stop` option.

- On UNIX: Go to *SAS-configuration-directory*`/Lev1/` `DeploymentTesterServer` and run the command `DeploymentTesterServer.sh -start`. When you are finished using the Deployment Tester, stop the server by using the same command with the `-stop` option.

- For z/OS, see "Starting and Stopping SAS Servers on z/OS" on page 80.

Note: After you add a new server to your configuration (or install software that includes one or more additional servers), be sure to restart the Deployment Tester Server. Otherwise, the Deployment Tester will not recognize the new servers.

Deployment Tester Test Suites: Descriptions and Requirements

Deployment Tester Test Suites and Their Requirements

The following table describes the Deployment Tester test suites that are installed with the SAS Intelligence Platform and indicates whether each test suite requires JUnit or special credentials.

Test Suite	Test Description	JUnit Required?	Required User Credentials
Content Server Tests	Runs basic unit tests for validating the SAS Content Server.	Yes	Unrestricted user account (such as sasadm@saspw)
OLAP Server Tests	Runs basic functionality tests for the SAS OLAP Server.	No	Unrestricted user account (such as sasadm@saspw) or an account that is a member of the SAS Administrators group
Pooled Workspace Server setinit Test	Creates a report of the SAS software that is installed on the SAS Pooled Workspace Server host machine. The report shows the license expiration date for each product.	No	Regular user account

Test Suite	Test Description	JUnit Required?	Required User Credentials
SAS OQ	Runs the SAS Operational Qualification Tool (SAS OQ) test suite, which demonstrates that your Base SAS software is operational.[*]	No	No credentials required
SAS IQ	Runs the SAS Installation Qualification Tool (SAS IQ) test suite, which demonstrates that your SAS software has been installed and maintained according to specifications.[*]	No	No credentials required
Stored Process Server Test	Runs basic tests for the SAS Stored Process Server.	Yes	Regular user account
Workspace Server setinit Test	Creates a report of the SAS software that is installed on the SAS Workspace Server host machine. The report shows the license expiration date for each product.	No	Regular user account (must have ReadMetadata permission on the server metadata). On Windows, the user might need to have a host account with rights to "Logon as a batch job."

* For details about SAS IQ and SAS OQ, see the *SAS Qualification Tools User's Guide* at http://support.sas.com.

Adding Tests to the Deployment Tester

Adding SAS OQ Tests

The primary input files for the SAS OQ test suite are test tables, which are stored in files with the extension .tab. These files define which tests to run, how to run the tests, and how to determine whether the tests worked correctly. You can add more SAS OQ tests to the test suite by creating new test tables or adding new tests to the existing tables. For details, see the *SAS Qualification Tools User's Guide* at http://support.sas.com.

Adding the AIX System Tuning Test Suite

Administrators can use the AIX System Tuning test suite for SAS Deployment Tester to optimize the performance of AIX environments. The test suite compares the AIX configuration to recommended best practices. To download the test suite, go to the Enterprise Management Integration page at http://support.sas.com/rnd/emi/DeployTest/index.html.

Using the Deployment Tester Interactively from SAS Management Console

Launching Deployment Tester Tests from SAS Management Console

To run the Deployment Tester interactively from SAS Management Console:

1. Start the Deployment Tester server on the host machine for each component that you want to test.

2. Log on to SAS Management Console as a user who is a member of the **Management Console: Advanced** role or the SAS Administrators group. On the **Plug-ins** tab, expand the **Application Manager** node and the **Deployment Tester** node.

3. In the Deployment Tester tree, locate the node for the Deployment Tester server where you want to run tests. The server is identified by the host machine name and the port (for example, server.mycompany.com:10021). Expand the server node to see the list of available tests, as shown in the following example:
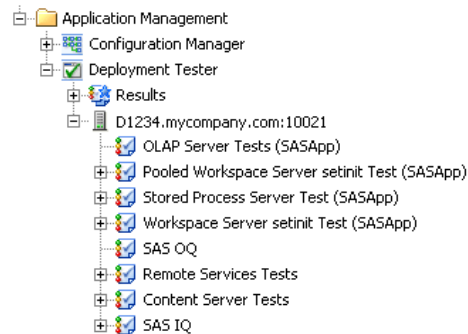
T I P If no tests appear, this might mean that the Deployment Tester server has not been started.

4. Right-click the test that you want to run, and select **Run Test Suite**. (Or if you want to run all of the tests that are available on this machine, right-click the node for the Deployment Tester server and select **Run All Test Suites**.)

5. Some tests require special user credentials. If you are prompted for credentials, enter the appropriate user ID and password. The password will be encrypted using SAS proprietary 32-bit (sas002) encryption. For details about required credentials, see "Deployment Tester Test Suites: Descriptions and Requirements" on page 100.

Test suites run as background processes, so you can perform other work while a test suite is running. When the tests are completed, the results are displayed in the right panel.

When you are finished executing tests, it is recommended that you stop the Deployment Tester server.

View Deployment Tester Results in SAS Management Console

When you run a Deployment Tester test suite interactively from SAS Management Console, the test results appear as soon as the test is completed. Each result is listed in the Results tree under the Deployment Tester plug-in. When you select a result in the

tree, the Test Suite Results report appears in the right pane, as shown in the following example:

A green icon with a check mark ( ) indicates that the test completed successfully.

CAUTION:
> **If you right-click a test result in the Results tree and select "Close Results," the result files are deleted from the tree and the associated documents are deleted from the file system.**

To view the output reports for a test, click the links on the results panel. Depending on which test you ran, you can view results by clicking the test name, **View stdout from test suite**, **View stderr from test suite**, or **View output files from test suite**.

If you need assistance in interpreting test results, contact SAS Technical Support.

Where Deployment Tester Result Files Are Stored When Tests Are Run from SAS Management Console

The result files that you can view from SAS Management Console are saved in the following paths in the file system:

- Windows Vista, Windows 7, and Windows Server 2008 systems: `C:\Users\`*user*`\AppData\Roaming\SAS\SASDeploymentTesterSASMCPlugin\2.2\Reports\`*yyyymmdd*`\report_`*test-suite-name*

- Other Windows systems: `C:\Documents and Settings\`*user*`\Application Data\SAS\SASDeploymentTesterSASMCPlugin\2.2\Reports\`*yyyymmdd*`\report_`*test-suite-name*

- UNIX: *user's-home-directory/*`SAS/SASDeploymentTesterSASMCPlugin/2.2/Reports/`*yyyymmdd*`/report_`*test-suite-name*

Note: When you right-click a test result in the Results tree and select **Close Results**, the result files are deleted from the tree and from the paths in the preceding list.

Permanently Saving Deployment Tester Test Results

To create a permanent record of tests that have been run, you can save the results to a location other than the user's home directory. To save the results of a test, right-click the test in the Results tree and make one of the following selections:

- Select **Save Results to Folder** if you want to save the results as a collection of loose files. In the dialog box that appears, navigate to the path where the results are to be saved. Then create a folder in which to store the files, select the new folder, and click **Save**.

- Select **Save Results as Zip File** if you want to save the results in a compressed file. This feature is helpful if you want to send the test results to another party such as SAS Technical Support.

  In the dialog box that appears, you might want to navigate to a different directory before you save the file. To do so, select **All Files** in the **Files of type** field. After navigating to the correct directory, change **Files of type** back to **Zip files (*.zip)**. Then enter a filename and click **Save**.

Deleting (Closing) Deployment Tester Test Results

Periodically, you might want to delete old Deployment Tester results from the Results tree in SAS Management Console and from your home directory. If there is a chance that you will need to refer to the results in the future, be sure to first save the results to another location, as described in the preceding topic.

To delete Deployment Tester results from the Results tree and from the users's home directory, select one or more results in the tree. Then right-click the mouse and select **Close Results**.

Using the Deployment Tester Batch Client

About Running the Deployment Tester in Batch Mode

The Deployment Tester can be run as a batch client, which enables you to automate the testing process. To regularly check on the health of your system, you can set up your scheduling system to run the batch client daily, weekly, or on any other schedule.

The batch client can be run from any machine where the Deployment Tester batch client is installed. Before running the batch client, be sure to start the Deployment Tester server on the host machine for each component that you want to test.

Syntax for the Batch Deployment Tester Command

bclient.sh | bclient.bat

    -contract *SAS-installation-directory*/SASDeploymentTesterClient/2.2/
      Contract/SASTestSuite_contract.xml
      -Vserver=*dt-server*
      -Vdir=*contract-file-dir*
      -VcontractFile=*contract-file-name*

Syntax Description for the Batch Deployment Tester Command

bclient.sh | bclient.bat
specifies the command for starting the batch Deployment Tester. Use -bclient.sh for UNIX or -bclient.bat for Windows.

Requirement	Required

-contract *SAS-installation-directory*/**SASDeploymentTesterClient/2.2/Contract/ SASTestSuite_contract.xml**

> specifies the name and location of the launcher contract file, SASTestSuite_contract.xml. This file calls other test-specific contract files. By default, the SAS Deployment Wizard installs this file in **`SAS-installation-directory`/SASDeploymentTesterClient/2.2/Contract**.
>
> In **`SAS-installation-directory`**, specify the path where your SAS software is installed on the local machine.

Requirement	Required
Tip	When running the Content Server Tests, SAS IQ, or SAS OQ on a UNIX environment, specify an absolute path for the location instead of a symbolic linked path.
Example	`-contract C:/Program Files/SASHome/ SASDeploymentTesterClient/2.2/Contract/ SASTestSuite_contract.xml`

-Vserver=*dt-server*

> specifies the host machine and port for the Deployment Tester server that is to run the test. Use the format *host-name:port*. The value that you specify depends on which SAS component you are testing. If you are running a test suite for a SAS server, then specify the SAS Deployment Server that is installed on the host machine for the server that you are testing. If you are running SAS OQ or SAS IQ, then specify the SAS Deployment Server on the machine that contains the SAS software that you want to test.
>
> If you are running only the SAS OQ or SAS IQ test suite, and if the SAS software that you are testing is on the same machine as the batch client, you can either omit this parameter or specify a value of **`server-local`**.

Requirement	This parameter is optional in the following situation: if you are running only the SAS OQ or SAS IQ test suite, and if the SAS software that you are testing is on the same machine as the batch client. In all other situations, the parameter is required.
Tip	The host machine names and ports for each instance of the Deployment Tester server are displayed under **Application Management ⇨ Deployment Tester** in the **Plug-ins** tab of SAS Management Console.
Example	`-Vserver=server.mycompany.com:10021`

-Vdir=*contract-file-dir*

> specifies the directory path for the contract file that is specified in **-Vserver=***contract-file-name*. The path varies depending on which test suite you are running. To determine the correct path, see .

Requirement	Required

Tip		When running the Content Server Tests, SAS IQ, or SAS OQ on a UNIX environment, specify an absolute path for the location instead of a symbolic linked path.

-VcontractFile=*contract-file-name*
specifies the name of the contract file for the test suite that you want to run. To determine the correct filename, see "Filenames and Locations for Deployment Tester Test Suite Contracts" on page 106.

Requirement	Required

-Vuserid=*my_userid*
specifies a valid user ID to access the component that is being tested.

Requirement	Required for the following test suites: Content Server Tests, OLAP Server Tests, Pooled Workspace Server setinit Test, Stored Process Server Test, and Workspace Server setinit Test.
See	For details about credential requirements, see "Deployment Tester Test Suites and Their Requirements" on page 100.

-Vpassword=*my_password*
specifies the password associated with the user ID that is specified in -Vuserid=. The password should be encrypted using SAS proprietary 32-bit encryption. To obtain the encrypted password, use PROC PWENCODE. If you do not encrypt the password, it will be transmitted using plain text.

Requirement	Required for the following test suites: Content Server Tests, OLAP Server Tests, Pooled Workspace Server setinit Test, Stored Process Server Test, and Workspace Server setinit Test.

Filenames and Locations for Deployment Tester Test Suite Contracts

The following table provides default directory paths and filenames for the contract files that are used for each Deployment Tester test suite. These values are used in the -Vdir= and -VcontractFile parameters of the batch command.

Note: The directory paths that are shown here are default locations. The paths in your deployment might be different.

Test Suite	Default Directory Path for Contract File for -Vdir=	Contract Filename for -VcontractFile
Content Server Tests	*SAS-configuration-directory*/Lev1/Web/Applications/ SASWIPServices9.4/dtest/SASContentServer	ContentServerTest_contract.xml
OLAP Server Tests	*SAS-configuration-directory*/Lev1/SASApp/OLAPServer/ dtest/OLAPServer	olap_contract.xml
Pooled Workspace Server setinit Test	*SAS-configuration-directory*/Lev1/SASApp/ PooledWorkspaceServer/dtest/PooledWorkspaceServer	SetinitTest_contract.xml

Test Suite	Default Directory Path for Contract File for -Vdir=	Contract Filename for -VcontractFile
SAS IQ	*SAS-installation-directory*/InstallMisc/utilities/ installqual/9.4	sasiq_contract.xml
SAS OQ	*SAS-installation-directory*/SASFoundation/9.4/dtest/ sasoq	sasoq_contract.xml
Stored Process Server Test	*SAS-configuration-directory*/Lev1/SASApp/ StoredProcessServer/dtest/StoredProcessServer	STPTest_2_contract.xml
Workspace Server setinit Test	*SAS-configuration-directory*/Lev1/SASApp/ WorkspaceServer/dtest/WorkspaceServer	SetinitTest_contract.xml

SAS Deployment Tester Batch Command: UNIX Example

The following is an example of a batch command issued on a UNIX machine to test a SAS OLAP Server that is located on a Windows machine:

```
cd $SASHOME/SASDeploymentTesterClient/2.2
bin/bclient.sh -contract Contract/SASTestSuite_contract.xml \
-Vserver="server.sas.com:10021" \
-Vdir "$SASConfig/Lev1/SASApp/OLAPServer/dtest/OLAPServer" \
-VcontractFile="olap_contract.xml" \
-Vuserid="user123" \
-Vpassword="{sas002}0123456789"
```

SAS Deployment Tester Batch Command: Windows Example

Here is an example of a batch command issued on a Windows machine to test a SAS OLAP Server that is located on a different Windows machine:

```
cd C:\Program Files\SASHome\SASDeploymentTesterClient\2.2
bin\bclient.bat -contract Contract\SASTestSuite_contract.xml
"-Vserver=server.sas.com:10021"
"-Vdir=C:\SAS\EBIServer\Lev1\SASApp\OLAPServer\dtest\OLAPServer"
-VcontractFile=olap_contract.xml
"-Vuserid=domain\user123"
"-Vpassword={sas002}0123456789"
```

Viewing Results of Tests That Are Run from the Deployment Tester Batch Client

If you run tests on a batch basis, you can access the test results by opening the following file:

- On Windows Vista, Windows 7, and Windows Server 2008 systems: **C:\Users *user*\AppData\Roaming\SAS\SASDeploymentTesterClient \2.2\Reports*yyyymmdd*\report_*test-suite-name*\index.html**

- On other Windows systems: `C:\Documents and Settings\`*`user`*`\Application Data\SAS\SASDeploymentTesterClient\2.2\Reports\`*`yyyymmdd`*`\report_`*`test-suite-name`*`\index.html`

- On UNIX: *`user's-home-directory`*`/.SASAppData/SASDeploymentTesterClient/2.2/Reports/`*`yyyymmdd`*`/report_`*`test-suite-name`*`/index.html`

Part 3

System Monitoring and Logging

Chapter 9

Monitoring the Activity of SAS Servers

About SAS Server Monitoring

To monitor the activity of SAS servers, including the metadata server, object spawners, OLAP servers, pooled workspace servers, SAS/CONNECT spawners, stored process servers, and workspace servers, you can use the following capabilities:

- the server management features of SAS Management Console. For details, see "Using SAS Management Console to Monitor SAS Servers" on page 112.

- the monitoring and management features of SAS Environment Manager. See "Using SAS Environment Manager to Monitor SAS Servers" on page 128.

- third-party monitoring tools. For details, see "Using Third-Party Products to Monitor SAS Servers" on page 128.

To monitor and manage SAS/SHARE servers, see the *SAS/SHARE User's Guide*.

Using SAS Management Console to Monitor SAS Servers

About SAS Management Console's Server Management Features

You can use SAS Management Console to monitor the activity of SAS servers, including the metadata server, object spawners, OLAP servers, pooled workspace servers, SAS/CONNECT spawners, stored process servers, and workspace servers. You can view the following types of information:

- currently connected clients

- graphs showing the number of spawned server sessions over time

- sessions that are active, inactive, or terminated

- server options and properties

- performance counters

- loggers and associated logging levels

- logging messages, at the level that you specify

This information is displayed on tabs in SAS Management Console, as shown in the following example:
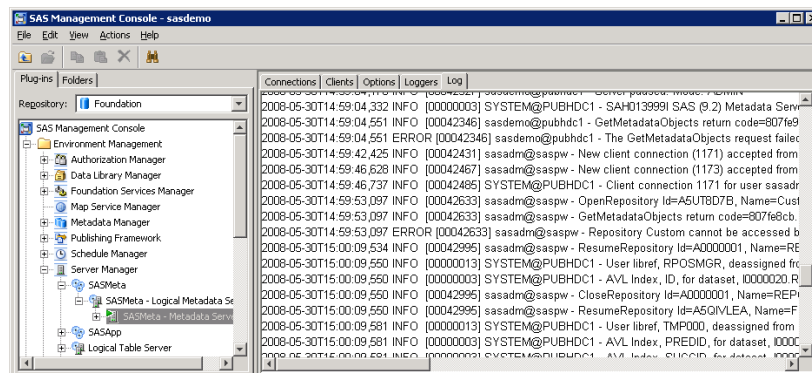
You can also use the server management features to dynamically change logging levels and some server properties.

Summary of Steps for Using SAS Management Console Server Monitoring

To use the server management features of SAS Management Console, follow these steps:

1. Obtain the appropriate user credentials and clear cached credentials (if necessary).

2. Use SAS Management Console to connect to servers and display server monitoring information.

3. Use the Server Manager tabs to obtain monitoring information and to adjust logging levels (if needed).

4. Return logging to normal levels, disconnect from the servers, and stop displaying log messages.

Each of these steps is described in the following topics.

Obtain the Appropriate User Credentials, and Clear Cached Credentials (If Necessary)

To perform server monitoring in SAS Management Console, you must have the appropriate user credentials. To determine what credentials are required for each group of tasks, see "Who Can Do What in the Server Manager Plug-in" on page 41.

If you attempt to connect to a server or spawner and you are denied access, make sure you have the appropriate credentials. After your credentials are updated, be sure to clear the credentials cache before you try to connect again. To clear the credentials cache, click **File ⇨ Clear Credentials Cache** in SAS Management Console.

Use SAS Management Console to Connect to Servers and Display Server Monitoring Information

To monitor an instance of a server or spawner, you must establish a connection. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Server Manager Plug-in" on page 41.

2. On the **Plug-ins** tab, expand the **Server Manager** node and the application server node (for example, **SASApp** or **SASMeta**).

3. Expand the node for the logical server that you want to monitor.

   The instances of that server that have been defined are displayed in the server tree. When you highlight a server instance in the tree, the right pane displays the available connections, as well as the port and the host machine for each connection.

4. To connect to the metadata server or an OLAP server and display monitoring information:

   a. In the Server Manager tree, right-click the server instance that you want to monitor.

   b. Click **Connect**.

      The following server monitoring tabs become active: **Clients**, **Options**, **Loggers**, and **Log**. For OLAP servers, a **Sessions** tab also appears.

5. To connect to a workspace server, pooled workspace server, or stored process server and display monitoring information, follow these steps:

   a. In the Server Manager tree, expand the node for the server instance that you want to monitor.

      The server's available connections are displayed in the server tree and in the right pane.

   b. In the Server Manager tree, highlight a connection and click **Connect**.

For each process that is active for that connection, the right pane displays the process ID, the process owner, and the port, as shown in this example:

Processes		
ID	Owner	Port
3772	sassrv@D1234	8611

c. In the Server Manager tree, expand the node for the connection that you are working with.

d. In the Server Manager tree, highlight the process that you want to monitor, as shown in this example:
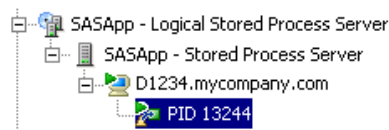
The following server monitoring tabs become active: **Clients**, **Options**, and **Loggers**. For stored process servers, the **Sessions** and **Log** tabs also become active.

6. To connect to an instance of the object spawner and display monitoring information, follow these steps:

a. On the **Plug-ins** tab, expand the **Server Manager** node and the **Object Spawner** node.

b. Right-click the object spawner instance that you want to administer, and click **Connect**.

The following server monitoring tabs become active: **Clients**, **Spawned Server Activity**, **Options**, **Loggers**, and **Log**.

7. To connect to an instance of the SAS/CONNECT spawner and display monitoring information, follow these steps:

a. On the **Plug-ins** tab, expand the **Server Manager**.

b. Right-click the **Connect Spawner** node and click **Connect**.

The following server monitoring tabs become active: **Connections**, **Clients**, **Spawned Server Activity**, **Spawned Servers**, **Options**, **Loggers**, and **Log**.

Note: If some tabs are disabled, then you need additional user credentials. See "Who Can Do What in the Server Manager Plug-in" on page 41.

To use the information on the tabs, see "Use the Server Manager Tabs to Obtain Monitoring Information and to Adjust Logging Levels (If Needed)" on page 115.

Use the Server Manager Tabs to Obtain Monitoring Information and to Adjust Logging Levels (If Needed)

About the Server Manager Tabs

After you have set the Server Manager options and connected to a server or a spawner, as described in the preceding topics, SAS Management Console displays server monitoring information in the right pane in a series of tabs.

Depending on which server component you are monitoring, SAS Management Console might display the following tabs:

- Clients

- Sessions

- Spawned Server Activity

- Options

- Loggers

- Log

Note: If some tabs are disabled, then you might need to connect to the server or you might need additional user credentials. See "Obtain the Appropriate User Credentials, and Clear Cached Credentials (If Necessary)" on page 113.

Use the Clients Tab in the Server Manager

The **Clients** tab in the Server Manager lists the clients that are or have been connected to the currently selected server component. The tab displays each client's user name, host machine, and entry time, as shown in the following example:

User	Host	Entry Time
sastrust@saspw	10.16.14.117:1747	May 30, 2008 10:00:07 AM
sastrust@saspw	10.16.14.117:1698	May 30, 2008 9:56:07 AM
sastrust@saspw	10.16.14.117:1971	May 30, 2008 10:15:23 AM
sastrust@saspw	10.16.14.117:2004	May 30, 2008 10:16:18 AM
sastrust@saspw	10.16.14.117:1949	May 30, 2008 10:14:46 AM
sasadm@saspw	10.16.14.117:1696	May 30, 2008 9:56:04 AM
sasdemo@pubhdc1	10.16.14.117:2934	May 30, 2008 2:01:17 PM
sastrust@saspw	10.16.14.117:1934	May 30, 2008 10:13:00 AM
sasdemo@PUBHDC1	10.16.14.117:2081	May 30, 2008 10:24:45 AM

The **Clients** tab is displayed for the following server components:

- metadata server

- object spawner connections

- OLAP servers

- pooled workspace server processes

- SAS/CONNECT spawners

- stored process server processes

- workspace server processes

Use the Sessions Tab in the Server Manager

The **Sessions** tab in the Server Manager lists the sessions that are active on the currently selected server or server process. Depending on your options settings, inactive and terminated sessions might also be displayed, as shown in this example:

To specify which types of sessions to display:

1. Right-click the **Server Manager** node, and select **Options**. The Options dialog box appears.

2. On the **Sessions** tab of the Options dialog box, select one of the following options:

   Active and Inactive
   > to display only active and inactive sessions in the **Sessions** tab. When sessions terminate, they will be removed from the tab.

   Active, Inactive, and Ended
   > to display active, inactive, and terminated sessions in the **Sessions** tab.

If terminated sessions are displayed, you can remove them by right-clicking the selected server node or process node in the server tree and selecting **Refresh**.

The **Sessions** tab is displayed for the following server components:

• OLAP servers

• stored process server processes

Use the Spawned Server Activity Tab in the Server Manager

The **Spawned Server Activity** tab in the Server Manager displays a graph that shows server activity associated with the currently selected SAS/CONNECT spawner or object spawner connection. For example, the graph for the object spawner shows how many spawned workspace servers, pooled workspace servers, and stored process servers are active at a given point in time, as shown in the following example:
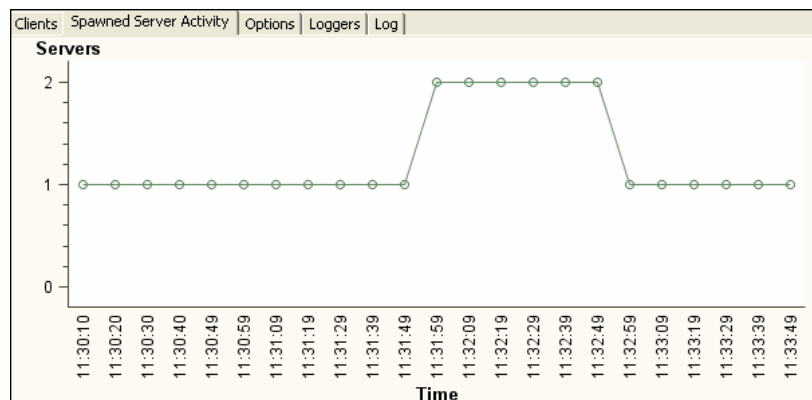
The graph is updated every 10 seconds. This tab is displayed only for object spawner connections and SAS/CONNECT spawners.

Use the Options Tab in the Server Manager

The **Options** tab displays performance counters and other information fields for the currently selected server or spawner session, as shown in the following example:

For more information about the performance counters and information fields, see "Using the Server Performance Counters and Information Fields" on page 121.

Some types of properties (for example, the JNLLINEMAX and JNLSTRMAX object server parameters) can be modified in this tab. To change the value of a property, right-click the property entry and click **Edit**. The changes remain in effect until the server component or process stops. When the server component is restarted, the options revert to those that are defined in the server's configuration file.

On the standard workspace servers, you can use information about this tab to identify orphaned server processes and determine the appropriate action to take. See "Using Performance Counters and Information Fields to Detect Orphaned Workspace Server Processes" on page 126.

Effective with SAS 9.4, performance counters can be reset. See "Resetting Performance Counters" on page 125.

The **Options** tab is displayed for the following server components:

- metadata server
- object spawner connections
- OLAP servers
- pooled workspace server processes
- SAS/CONNECT spawners
- stored process server processes
- workspace server processes

Use the Loggers Tab in the Server Manager, and (If Necessary) Change Logging Levels for Individual Loggers

The **Loggers** tab in the Server Manager displays the loggers that are in use for the selected server component, as shown in the following example:

Connections	Clients	Options	Loggers	Log

Name △	Description	Level
`<Root>`		Error
Admin		Information
Admin.Operations		`<Inherited>`
App		Information
App.FilteredList		`<Inherited>`
App.FilteredList.FilteredList		`<Inherited>`
App.FilteredList.FilteredList.Close		`<Inherited>`
App.FilteredList.FilteredList.Filter		`<Inherited>`
App.FilteredList.FilteredList.Filter.Get		`<Inherited>`
App.FilteredList.FilteredList.Filter.Set		`<Inherited>`
App.FilteredList.FilteredList.GetAttrib...		`<Inherited>`

For each logger that is associated with the server component, this tab shows the logging level that is in effect. If **<Inherited>** appears in the third column, then the logging level is not set explicitly for the logger. Instead, the level is inherited from a parent logger. If you want to see which logging level is in effect through inheritance, then right-click the entry and select **Properties**.

The logging levels that are in effect at server start-up are based on the server's logging configuration file. In most cases, there is no need to change the logging levels. However, SAS Technical Support might request that you change a logging level temporarily to assist in problem resolution. For example, you might be requested to increase the level for the object spawner's `App` logger.

To change the logging level temporarily for a logger:

1. Right-click the entry for the logger whose level you want to change, and select **Properties**.

2. In the **Properties** dialog box, select the **Assigned** radio button. Then select the appropriate level from the drop-down box.

CAUTION:
> **Excessive logging can degrade performance. You should not use the TRACE and DEBUG logging levels unless you are directed to do so by SAS Technical Support.**

The logging level changes that you enter here have the following effects on logging:

• For the duration of the server process, the changes override the logging levels that are defined in the server's logging configuration file.

• The changes affect messages that are displayed in the **Log** tab, as well as messages that are sent to other appenders (as configured in the server's logging configuration file).

• The changes remain in effect until the server component or process stops. When the server component is restarted, the logging levels revert to those that are defined in the server's logging configuration file.

Note: The logging levels that are set in the default configuration files should be sufficient for most sites. If you need to permanently change logging levels for a component, then you must change the logging configuration file. To maintain system performance, use extreme caution in making these changes.

The **Loggers** tab is displayed for the following server components:

• metadata server

• object spawner connections

- OLAP servers

- pooled workspace server processes

- SAS/CONNECT spawners

- stored process server processes

- workspace server processes

For more information, see Chapter 10, "Administering Logging for SAS Servers," on page 131 in this guide, as well as the *SAS Logging: Configuration and Programming Reference*.

Use the Log Tab in the Server Manager

The **Log** tab displays log events for the currently selected server or server process, as shown in the following example:

To display log events on this tab:

1. Use the Server Manager options to specify a message level or threshold level filter. You can also specify a starting and ending time.

   Note: These settings filter the events that are already being generated based on the server's logging configuration file and the **Loggers** tab; they do not generate additional events. If you want to generate additional events, use the **Loggers** tab to change the level for one or more loggers, or modify the server's logging configuration file.

   Set the filters as follows:

   a. Right-click the **Server Manager** node, and select **Options**. The Options dialog box appears.

   b. Click the **Logging** tab of the Options dialog box, and specify the following options:

      Message Level
      specifies a specific level of messages that are to be displayed in SAS Management Console.

      Threshold Level
      specifies the lowest level of messages that are to be displayed in SAS Management Console. Messages that have a level equal to or higher than the specified level are displayed in the **Log** tab.

      Note: You can specify either a threshold level or a message level, but you cannot specify both.

Start Time
specifies the date and time that log messages are to start appearing in SAS Management Console. Click the date and time icon (🗓), specify a date

and time, and then click **OK** to save your entry.

If you do not specify a start time, then the message level or threshold level takes effect immediately.

End Time
specifies the date and time that log messages are to stop appearing in SAS Management Console. Click the date and time icon (🗓), specify a date

and time, and then click **OK** to save your entry.

If you do not specify an end time, then the message level or threshold level remains in effect until you change it.

Click **OK** in the Options dialog box.

2. Click **View** ⇨ **Refresh** on the menu bar to display the current contents of the message buffer.

3. Click **View** ⇨ **Refresh** again each time you want to see the latest events.

The events that are displayed in this tab depend on the following:

• the loggers and logging levels that are specified in the server's logging configuration file.

• adjustments to logging levels that you might have made on the Loggers tab. These adjustments override the logging configuration file and remain in effect for the duration of the server session.

• additional filtering (based on message level, threshold level, and starting and ending times) that you specify in the Server Manager options.

For example, if you specify a threshold level of TRACE in the Server Manager Options, but none of the loggers are set to TRACE in the logging configuration file or on the **Loggers** tab, then no TRACE events will appear in SAS Management Console. You must first adjust the logging level for the appropriate loggers either in the logging configuration file or the **Loggers** tab.

CAUTION:
Excessive logging can degrade performance. Therefore, you should not use the TRACE and DEBUG logging levels unless you are directed to do so by SAS Technical Support.

The **Log** tab is displayed for the following server components:

• metadata server

• object spawner connections

• OLAP servers

• SAS/CONNECT spawners

• stored process server processes

For more information, see Chapter 10, "Administering Logging for SAS Servers," on page 131 in this guide, as well as the *SAS Logging: Configuration and Programming Reference*.

Return Logging to Normal Levels, Disconnect from the Servers, and Stop Displaying Log Messages

When you are finished monitoring servers, you should do the following:

1. If you changed the logging levels using the Loggers tab,, then manually reset the loggers to their normal levels.

   Note: You can also reset loggers to their normal levels by restarting the server component.

2. Reset the message level, threshold level, or the start and end times in the Server Manager options.

3. Disconnect from any server components that you are connected to.

Using the Server Performance Counters and Information Fields

About the Server Performance Counters and Information Fields

Various types of performance counters and information fields are provided for the following:

- the metadata server

- instances of OLAP servers, object spawners, and SAS/CONNECT spawners

- active sessions on pooled workspace servers and stored process servers

The performance counters provide statistics about activity that has occurred since the server or spawner was last started. The information fields provide static information about the server or spawner configuration or about the current server or spawner session.

The description and the current value of each performance counter and information field are displayed on the server's **Options** tab, which appears on the right pane of SAS Management Console. To access this tab, follow the instructions in "Using SAS Management Console to Monitor SAS Servers" on page 112.

Effective with SAS 9.4, you can use SAS Management Console to reset performance counters. See "Resetting Performance Counters" on page 125.

You can also use third-party monitoring tools to view the counters and information fields and to reset the performance counters. See "Using Third-Party Products to Monitor SAS Servers" on page 128.

The following types of performance counters and information fields are provided:

CNTSPAWN.*information-type*
   provides status and static information about the selected SAS/CONNECT spawner.

IOM.*information-type*
   provides status and static information about the selected metadata server, OLAP server, object spawner, SAS/CONNECT spawner, pooled workspace server, or stored process server session, including the following information:

   - the server or server start time

- the current execution state

- the port on which the server or spawner is listening

- the time at which the counters were last reset

- the name of the server or spawner component

`IOM.Counter`*counter-name*
records activity for the selected metadata server, OLAP server, object spawner, pooled workspace server, or stored process server session, including the following information:

- the number of times that the counters were reset

- the number of currently connected clients

- the amount of time that the server or spawner has been idle

- the amount of time that has been spent processing server method calls

- the number of method calls that have been processed

- the number of clients that have been serviced

`LogicalServer.`*logical-server-name***.***counter-name*
records object spawner activity for each logical server that is serviced by the object spawner. This type of counter appears in the object spawner's **Options** tab.

For logical pooled workspace servers and stored process servers, these counters provide the following information:

- the number of clients that are connected, pending, and waiting

- the number of clients that have been redirected, that have timed out, and that have been denied authorization

- the number of servers and the number of active peers that are running in the cluster

- the total waiting time for clients

For logical pooled workspace servers, the counters provide information about the total waiting time for clients and the number of clients that have been denied authorization. For logical non-pooled workspace servers, a counter provides the number of clients that have been denied authorization.

`LogicalServer.`*logical-server-name***.***information-type*
provides static information about each logical server that is serviced by the object spawner. This type of field appears in the object spawner's **Options** tab. These fields provide information such as the logical server's class identifier, authorization service, security settings, maximum wait time, and load balancing configuration (if applicable).

`LogicalServerName.`*n*
provides the names of each logical server that is serviced by the object spawner. This field appears in the object spawner's **Options** tab.

`Metadata.Counter`*method-name*
records the number of times that specific metadata server methods are called. This type of counter provides information such as the following:

- the number of times that SAS Data Integration Studio calls the change management methods that are used to check out and check in metadata

- the number of times that the metadata server is paused and resumed

Metadata.CounterIO*transaction-type*
> records low-level input and output activity on metadata repository data sets. This type of counter is updated as follows:
>
> * Each time a repository, metadata object, or association is created, deleted, updated, or read in any metadata repository on the server, the appropriate counter is incremented by one.
>
> * If a metadata object is the first object of its type to have an operation performed, then counters are incremented by more than one. For example, if an object is added and the object's metadata type does not yet exist in the repository, then Metadata.CounterIOAdd is incremented by 3.
>
> * When a repository is deleted, the Metadata.CounterIODelete counter is incremented by one and the Metadata.CounterIOAdd counter is incremented by one because a new container registry is created.

Metadata.CounterJournal*counter-name*
> records activity and status information for the metadata server's journal file. This type includes the following counters:
>
> **Metadata.CounterJournalDataAvailable**
>> indicates the number of bytes of data that are in the journal file that have not yet been applied to the repository data sets on disk.
>
> **Metadata.CounterJournalDecompDataAvailable**
>> provides the raw number of bytes for the Metadata.CounterJournalDataAvailable counter, before compression.
>
> **Metadata.CounterJournalDataAvailableMax**
>> indicates the largest amount of metadata, in bytes, that has been committed to memory but not to repository data sets on disk. This value is equal to the highest value that has been placed in the Metadata.CounterJournalDataAvailable counter during the current server session. You can use this counter to determine the appropriate value for the JOURNALFILESIZE parameter of the omaconfig.xml file. If the omaconfig.xml JOURNALFILESIZE option specifies a journal file size that is too small, the metadata server slows down to allow the background task to catch up.
>
> **Metadata.CounterJournalDecompDataAvailableMax**
>> provides the raw number of bytes for the Metadata.CounterJournalDataAvailableMax counter, before compression.
>
> **Metadata.CounterJournalDataWritten**
>> indicates the total number of bytes of metadata that have been written to the journal file since the file was created. This number continues to grow throughout the life of the journal file and is not reset when the server is stopped and restarted.
>
> **Metadata.CounterJournalDecompDataWritten**
>> provides the raw number of bytes for the Metadata.CounterJournalDataWritten counter, before compression.
>
> **Metadata.CounterJournalMaxFileEntrySize**
>> indicates the maximum number of bytes of data in a single journal entry in the journal.
>
> **Metadata.CounterJournalMaxDecompEntrySize**
>> provides the raw number of bytes for the Metadata.CounterJournalMaxFileEntrySize counter, before compression.

Metadata.CounterJournalSpaceAvailable
indicates either the number of bytes of space left in the journal file (if the file is a fixed size), or 999999999 if the journal file does not have a fixed size.

Metadata.CounterJournalTransProcessed
indicates the number of transactions that have been fully processed to the data sets on disk since the metadata server was last started.

Metadata.CounterJournalTransQueued
indicates the number of transactions in the journal file that are currently waiting to be applied to metadata repositories on disk. This counter corresponds to the Metadata.CounterJournalDataAvailable counter, which measures the outstanding workload in bytes.

Metadata.CounterJournalTransWritten
indicates the number of transactions that have been processed since the server was started with journaling enabled. The number includes transactions that are still in the journal file and those that have been applied to repositories on disk. This number is reset when the server is paused to an Offline state and resumed, and also when the server is stopped and restarted.

Metadata.CounterRetryUpdateLock*counter-name*
provides information to help diagnose performance issues with metadata updates that involve multiple repositories, particularly if messages indicate that repositories were not properly locked. This type includes the following counters:

Metadata.CounterRetryUpdateLockMiss
indicates the number of missing repository update locks that have been encountered.

Metadata.CounterRetryUpdateLockMissCount
indicates the number of times that requests were restarted because a repository update lock was missing.

Metadata.CounterRetryUpdateLockRequest
indicates the number of requests that were restarted because a repository update lock was missing.

Metadata.CounterRetryUpdateLockHighWater
indicates the highest number of update lock retries that occurred for a single request. This value cannot exceed the update retry limit, which is specified in the omaconfig.xml file's MAXIMUM_RETRY_COUNT option.

Metadata.CounterRetryUpdateLockFailed
indicates the number of requests that could not be completed within the update retry limit, which is specified in the omaconfig.xml file's MAXIMUM_RETRY_COUNT option.

Metadata.CounterRetryUpdateLockExtra
indicates the number of repository update locks that have been obtained but not used. This number includes update locks for requests that only needed to read information from the repository.

Metadata.CounterRetryUpdateLockExtraCount
indicates the number of requests that obtained but did not use an update lock. This includes requests that only needed to read information from the repository.

For more information about repository update locks, see

Metadata.CounterRetryQueryLock*counter-name*
   provides information to help diagnose performance issues with metadata queries that involve multiple repositories, particularly if messages indicate that repositories were not properly locked. This type includes the following counters:

Metadata.CounterRetryQueryLock
   indicates the number of query locks that were needed after requests were started.

Metadata.CounterRetryQueryLockMiss
   indicates the number of missing repository query locks that could not be obtained, and the number of times that query requests were restarted because of missing query locks.

Metadata.CounterRetryQueryLockRequest
   indicates the number of query requests that were restarted because query locks could not be obtained.

Metadata.CounterRetryQueryLockHighWater
   indicates the highest number of query lock retries that occurred for a single request. This value cannot exceed the query retry limit, which is specified in the omaconfig.xml file's MAXIMUM_QUERY_RETRY_COUNT option.

Metadata.CounterRetryQueryLockFailed
   indicates the number of query requests that could not be completed within the query retry limit, which is specified in the omaconfig.xml file's MAXIMUM_QUERY_RETRY_COUNT option. The default for this option is 7 or the number of repositories that were registered when the server was last started, whichever is greater. If the Metadata.CounterRetryQueryLockFailed counter is greater than zero, and if there is a possibility that new repositories were created since the server was last started, then you might want to restart the metadata server in order to reset the default.

For more information about repository query locks, see

OBJSPAWN.*counter-name*
   records activity for the selected object spawner session, including the number of servers launched, the number of servers abandoned, and the number of servers that failed to launch.

Server.*information-type*
   provides static information about the selected metadata server, OLAP server, object spawner, SAS/CONNECT spawner, pooled workspace server, or stored process server session, including the following information:

   • the number of CPUs that are available

   • the executable that was used to launch the server or spawner

   • the identity under which the server or spawner process is executing

   • the process ID

   • the name of the computer where the server or spawner process is executing

Resetting Performance Counters

To reset the performance counters for the metadata server, an OLAP server, or an object spawner, follow these steps:

1. In the Server Manager tree, right-click the server or spawner instance whose counters you want to reset, and click **Connect**.

2. Right-click the server instance again, and click **Reset Counters**.

To reset the performance counters for a pooled workspace server session or a stored process server session, follow these steps:

1. In the Server Manager tree, expand the node for the server instance whose counters you want to reset. The server's available connections are displayed in the server tree.

2. In the Server Manager tree, highlight a connection and click **Connect**.

3. Expand the node for the connection that you are working with. The active processes for the connection are displayed in the tree.

4. Right-click the process whose counters you want to reset, and click **Reset Counters**.

Note: Counters for the SAS/CONNECT spawner cannot be reset.

Using Performance Counters and Information Fields to Detect Orphaned Workspace Server Processes

On the SAS Workspace Server, you can use the IOM.CurrentClients counter and the IOM.ReconnectEnabled field to identify orphaned processes and determine the appropriate action to take.

If IOM.CurrentClients is equal to 1, then the server has a client and the server process should not be stopped.

If IOM.CurrentClients is equal to 0, then the server has lost the connection with the client. To determine how to proceed, check the IOM.ReconnectEnabled setting:

• If IOM.ReconnectEnabled is set to true, the server has been requested to allow a previous client to reconnect. Therefore, you might want to keep the server process active in case the client reconnects.

• If IOM.ReconnectEnabled is set to false, the server will not allow the previous client to reconnect. By default, the server detects that the process is orphaned and stops it automatically.

This default behavior is controlled by the **Shut down processes without clients** option in SAS Management Console. If this option has been deselected (or if the WKSP_KILL_OFF environment variable is in effect), then you need to stop orphaned server processes manually. You can do so by right-clicking the process node in SAS Management Console and selecting **Stop**.

Note: This information applies only to standard workspace servers. It does not apply to pooled workspace servers or stored process servers.

Using the SAS OLAP Server Monitor

Access the SAS OLAP Server Monitor

The SAS OLAP Server Monitor in SAS Management Console enables you to perform the following tasks:

• connect to SAS OLAP Servers

• refresh the server list, the connections list, and cube metadata

- monitor sessions and queries

- close sessions

- stop SAS OLAP Servers

Follow these steps to access the SAS OLAP Server Monitor:

1. Start SAS Management Console. If necessary, choose a metadata profile that has the Administer permission and the Advanced role.

2. In the navigation tree, expand the **Monitoring** node and the **SAS OLAP Server Monitor** node.

3. Right-click the name of the logical OLAP server, and select **Connect**. Enter your user ID and password at the prompt.

4. When you are connected, right-click the logical server name and select an action.

5. When you are finished using the monitor, right-click the name of the logical OLAP server and select **Disconnect**.

Note: Online Help becomes available when you select SAS OLAP Server Monitor in the navigation tree.

View OLAP Server Sessions and Queries

To view the active SAS OLAP Server sessions and queries, follow these steps:

1. Access the SAS OLAP Server Monitor.

2. Under **Monitoring** and **SAS OLAP Server Monitor**, expand the **Logical OLAP Server** node and double–click the **OLAP Server**. The right pane displays a list of active OLAP sessions. In the Session Owner column, if the server is not secured, the owner is displayed as `<unknown>`.

3. To list the active queries for a session, double-click the session in the tree. For each active query, SAS Management Console displays the type of query (unspecified, multidimensional, or flattened) and the MDX expression.

Refresh the List of OLAP Servers or Server Connections

After you have accessed the SAS OLAP Server Monitor,, you can do the following:

- To refresh the list of SAS OLAP Servers, right-click **SAS OLAP Server Monitor** and select **Refresh**.

- To refresh the list of connections to a SAS OLAP Server, expand the **Server Manager** node, the SAS Application Server node (for example, **SASMain**), and the **Logical OLAP Server** node in the navigation tree. Then right-click the OLAP server name, and select **Refresh**.

Note: For information about refreshing the cube metadata for one or more cubes in the SAS OLAP Server's cube cache, see the *SAS Intelligence Platform: Application Server Administration Guide*.

Using SAS Environment Manager to Monitor SAS Servers

SAS Environment Manager is a web-based application with robust features to help you monitor and manage the availability and overall health of the servers and services in your SAS environment. A variety of metric data, in both numeric and graphic format, provides detailed information about the operation of each resource. For example, you can view the number of clients for each resource, as well as data about CPU and memory usage. In addition, you can configure SAS Environment Manager to monitor directory space usage, log file events, and configuration file changes and to report alerts.

To view monitoring information in SAS Environment Manager, select the **Resources** tab and then select the resource that you want to monitor.

Because SAS Environment Manager is based on Hyperic from VMWare, some server names in SAS Environment Manager do not match the names that are used in a SAS deployment. Use this table to determine the name of the server that you are interested in:

SAS Server Name	Name Displayed in SAS Environment Manager
SAS Environment Manager	Apache Tomcat 5.5, 6.0, 7.0
SAS Environment Manager	HQ Agent
SAS Web Server	vFabric Web Server 5.1, 5.2
SAS Web Application Server	SpringSource tc Runtime 7.0
SAS Web Infrastructure Platform Data Server	PostgreSQL 9.x HQ PostgreSQL 9.x vPostgreSQL 9.x
SAS JMS Broker	Active MQ 4.0, 5.0, 5.1, 5.3, 5.3, 5.4, 5.7

Detailed instructions for these monitoring features are provided in the *SAS Environment Manager User's Guide* and in the Help that is available from within the application.

Using Third-Party Products to Monitor SAS Servers

In addition to using SAS Management Console and SAS Environment Manager to monitor SAS servers, you can use the following enterprise systems management products:

• BMC Performance Manager (formerly BMC PATROL)

• Boemska Enterprise Session Monitor for SAS

- HP Insight Control Environment (ICE)

- HP OpenView

- Hyperic HQ

- IBM Tivoli

- Xymon

For information about how to enable the use of these products with the SAS Intelligence Platform, see **http://support.sas.com/rnd/emi**.

In addition, SAS provides MBeans that can be used to manage object spawners, metadata servers, workspace servers, pooled workspace servers, stored process servers, and OLAP servers. For details, see "Use JMX Tools to Manage SAS Resources " in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Chapter 10

Administering Logging for SAS Servers

About Server Logging

The SAS Intelligence Platform uses a standard logging facility to perform logging for SAS servers. The facility supports problem diagnosis and resolution, performance and capacity management, and auditing and regulatory compliance. The logging facility provides the following capabilities for servers:

- Log events are categorized using a hierarchical naming system that enables you to configure logging at a broad or a fine-grained level.

- Log events can be directed to multiple output destinations, including files, operating system facilities, and client applications. For each output destination, you can specify the following logging facility components:

  - the categories and levels of log events to report

  - the message layout, including the types of data to be included, the order of the data, and the format of the data

  - filters based on criteria such as diagnostic levels and message content

- Performance-related log events can be generated for processing by the Application Response Measurement (ARM) 4.0 server.

- You can use SAS Management Console and third-party monitoring tools to view the contents of logs.

- You can adjust logging levels dynamically by using the Server Management features of SAS Management Console.

Each server has a logging configuration file that controls the destination, contents, and format of the log for that server. The configuration file specifies options for the following components of the logging facility:

appenders
> are named entities that represent a specific output destination for messages. Destinations include fixed files, rolling files, operating system facilities, and client applications. You can configure appenders by specifying thresholds, filters, log directories and filenames, pattern layouts, and other parameters that control how messages are written to the destination.

loggers
> are named entities that identify message categories. Loggers are named using a hierarchical system that enables you to configure logging at a broad or a fine-grained level.

> The logging facility includes a set of high-level loggers for SAS servers, including Audit, Admin, App, IOM, and Perf. Some loggers are subdivided into lower-level (child) loggers. For example, the Audit logger has descendant loggers called Audit.Meta and Audit.Authentication, and Audit.Meta has descendant loggers called Audit.Meta.Security, Audit.Meta.Mgmt, and Audit.Meta.Updates. The Root logger is the highest-level logger and does not represent a specific message category.

> Loggers inherit settings from their higher-level (ancestor) loggers.

pattern layouts
> are templates that you create to format messages. The pattern layout identifies the types of data, the order of the data, and the format of the data that is generated in a log event and is delivered as output.

For each logger and appender that you specify, you can specify a threshold level that limits the events that will be created or written. Logging levels, from lowest to highest, are ALL, TRACE, DEBUG, INFO, WARN, ERROR, and FATAL. For example:

- If you specify a threshold level of INFO for a logger, then the logger creates only the events with levels of INFO, WARN, ERROR, and FATAL.

- If you specify a threshold level of ERROR for an appender, then events are written to the appender only if they have a level of ERROR or FATAL.

The SAS Deployment Wizard provides default logging configuration files for your SAS servers. You can modify these files as needed to adjust your logging configuration. You can also adjust logging levels dynamically by using the Server Management features of SAS Management Console. See "Using SAS Management Console to Monitor SAS Servers" on page 112.

The following topics describe the initial logging configuration for SAS servers and provide high-level instructions for administering server logging. For detailed information about the logging facility, see the *SAS Logging: Configuration and Programming Reference*.

About Metadata Server Loggers

The logging facility includes loggers for the following categories of events that are specifically related to the SAS Metadata Server:

- security auditing events

- metadata server and metadata repository management events

- change management events

- input and output events

- performance-related events

The initial logging configuration is set up to log these event categories, except for performance-related events, at the INFO, WARN, ERROR and FATAL levels. The loggers are as follows:

App.Meta
is the parent logger for metadata server events. Logging levels that are defined for this logger are inherited by its child loggers (including App.Meta.CM, App.Meta.IO, and App.Meta.Mgmt) unless they are explicitly overridden.

App.Meta.CM
logs change management events including check-in and check-out.

App.Meta.IO
logs low-level input and output activity.

App.Meta.Mgmt
logs metadata server management activity such as the following:

- server operation actions such as pause, resume, and refresh

- creating and deleting repositories

- modifying repository access modes

- repository backup and migration

If you want to capture these events in the operating system facility, see "Capture Metadata Server Management Events in the System Facility Appender" on page 147.

Audit.Meta.Security
> is the parent logger for metadata server security events. No events are written directly to this logger. Logging levels that are defined for this logger are inherited by its child loggers (for example, Audit.Meta.Security.AccCtrlAdm) unless the levels are explicitly overridden.

Audit.Meta.Security.AccCtrlAdm
> logs changes to permissions, permission settings, and access control templates (ACTs); and changes to passwords for objects such as tables and connections (including adding, deleting, and modifying passwords, as well as failed attempts to perform these actions).

Audit.Meta.Security.GrpAdm
> logs changes to group or role membership (including adding and removing members, and failed attempts to perform these actions).

Audit.Meta.Security.UserAdm
> logs changes to users, groups, roles, logins, and authentication domains (including adding, deleting, and modifying these objects, as well as failed attempts to perform these actions).

Audit.Meta.Updates
> is the parent logger for events that are performed on public objects. The logged events include adding, updating, deleting, checking out, checking in, and fetching objects, as well as undoing checkouts.
>
> Audit.Meta.Updates.PublicObjects logs events about public objects in foundation and custom repositories. Audit.Meta.Updates.Project and Audit.Meta.Updates.Project.PublicObjects log information about public objects in project repositories.

Perf.Meta
> logs performance-related events. This information previously was available through the OMA_ARM subsystem.

Perf.Meta.Expensive
> (new with SAS 9.4) logs requests that take longer than a specified time threshold so that application developers and administrators can identify high-cost metadata requests. The performance threshold is 30 seconds.

Here are some examples of security auditing messages.

```
2008-07-22T14:22:14,222 INFO  [00222108] 2504:User123@saspw - Added
IdentityType=Person  Name=Test User, ObjId=A5LL3EE0.AN00000D.

2008-07-22T14:22:14,222 INFO  [00222108] 2504:User123@saspw - Added Member

IdentityType=Person  Name=Test User, ObjId=A5LL3EE0.AN00000D to IdentityGroup
Name=BI Dashboard Users, ObjId=A5LL3EE0.A3000000.

2008-07-22T14:22:14,222 INFO  [00222108] 2504:User123@saspw - Added Login with

UserId=Test User, ObjId=A5LL3EE0.AQ00000F, AuthDomain=DefaultAuth to
IdentityType=Person Name=Test User, ObjId=A5LL3EE0.AN00000D.
```

These messages record the addition of a new user identity (Test User), the addition of the user to the BI Dashboard Users group, and the creation of a login for the user. Each message includes:

- the event date and time

- the logging level

- the thread identifier (in brackets)

- the client ID

- the user identity that is associated with the current thread or task

- the generated message, preceded by a hyphen

Note: The metadata server also generates performance counters. For details, see "Using the Server Performance Counters and Information Fields" on page 121.

Enabling Server Logging

The SAS Deployment Wizard automatically enables logging for most SAS servers. The following steps are necessary to enable server logging:

1. A logging configuration file must be created for the server. The logging configuration file is an XML file that specifies and configures loggers and appenders.

   The SAS Deployment Wizard automatically creates an initial logging configuration file for each server. Each server's configuration file is named logconfig.xml and is located in the directory where the servers' start-up script is located. For example, the metadata server's logging configuration file is located in the following path:

   SAS-configuration-directory/Lev1/SASMeta/MetadataServer

2. The LOGCONFIGLOC= system option must be specified in the server's sasv9.cfg file. This system option specifies the path and filename of the logging configuration file, as in the following example:

   -logconfigloc "logconfig.xml"

   If you want to change the option to specify a different path or filename for the logging configuration, then modify the LOGCONFIGLOC= option in the server's sasv9_usermods.cfg file.

For details about the initial configuration files, see "Initial Logging Configuration for SAS Servers" on page 135.

Initial Logging Configuration for SAS Servers

The following table shows the logging configuration that is created initially for each SAS server. For each server, the table displays the types of events that are written to each log destination.

Table 10.1 Initial Logging Configuration for SAS Servers

Server	Log Destination		
	Rolling Log File and SAS Management Console Log Tab	**Operating System Facility**	**ARM Logging Facility**
SAS Metadata Server	• Admin, App, Audit, and IOM events (INFO, WARN, ERROR, and FATAL) • Perf.Meta.Expensive events (INFO, WARN, ERROR, and FATAL) • all other events with a level of ERROR or FATAL	None	None
SAS object spawner	• App, Audit, and IOM events (INFO, WARN, ERROR, and FATAL) • all other events with a level of ERROR or FATAL	Admin events (INFO, WARN, ERROR, or FATAL)	None
SAS OLAP Server	• Admin events (ERROR and FATAL) • App, Audit, and IOM events (INFO, WARN, ERROR, and FATAL)	Admin.Operation events (ERROR and FATAL)	Performance-related events to support cube testing and tuning (captured in a log file if the logging level is modified). See "Collecting ARM Log Information in SAS OLAP Servers" on page 144.
SAS Pooled Workspace Server	• Admin events (ERROR and FATAL) • App, Audit, and IOM events (INFO, WARN, ERROR, and FATAL) • App.Program events (ERROR and FATAL) • all other events with a level of ERROR or FATAL Events are sent only to the rolling log file. The following loggers are set to immutable: Admin, App (except App.Program), IOM, and Audit.	None	None
SAS Stored Process Server	• Admin events (ERROR and FATAL) • App, Audit, and IOM events (INFO, WARN, ERROR, and FATAL) • App.Program events (FATAL) • all other events with a level of ERROR or FATAL The following loggers are set to immutable: Admin, App (except App.Program), IOM, and Audit.		

| | Log Destination | | |
Server	Rolling Log File and SAS Management Console Log Tab	Operating System Facility	ARM Logging Facility
SAS Workspace Servers	None	Admin.Operation events (ERROR and FATAL) The following loggers are set to immutable: Admin, App, IOM, and Audit.	Performance-related events (available for display in SAS Data Integration Studio)
SAS DATA step batch server	App events (INFO, WARN, ERROR, and FATAL)	Admin.Session events (ERROR and FATAL)	Performance-related events (captured in a log file if the server configuration is modified). See "Collecting ARM Log Information for SAS Data Integration Studio Jobs" on page 142.
SAS/CONNECT server	None	Admin.Session events (INFO, WARN, ERROR, and FATAL)	Performance-related events (available for display in SAS Data Integration Studio)

The log destinations that are shown in the preceding table are configured as follows:

Rolling log files

For metadata servers, object spawners, OLAP servers, pooled workspace servers, and stored process servers, the logging configuration file includes a definition of RollingFileAppender. This appender routes events to a rolling log file, which is configured as follows:

- A new log file is created when the date changes and when a new server or spawner process is started.

- Events are written using a layout that includes the following information:

  - the event date and time

  - the logging level

  - the thread identifier (in brackets)

  - the client ID

  - the user identity that is associated with the current thread or task

  - the generated message, preceded by a hyphen

  Here is an example of a metadata server event that was created with the specified layout:

  ```
  2008-10-15T11:42:49,294 INFO [00005671] User123@D1234 -
  Server paused. Mode: OFFLINE
  ```

- The rolling log files are named *server-type_current-date_process-id*.log.

For example, events for metadata server process 6188 on October 12, 2008, are written to a file called MetadataServer_2008-10-12_6188.log.

- As a default, the rolling log files are placed in the path ***server-configuration-directory*/Logs**. For example, log files for the metadata server are written to the following path:

  ***SAS-configuration-directory*/Lev1/SASMeta/MetadataServer/Logs**

  For a list of default log locations for servers and spawners, see "Default Locations for Server Logs" on page 445.

  Note: In a custom installation, you can choose to place all of your log files in a single directory. If this option was selected, then your log files are written to the path that was specified during installation (for example, ***SAS-configuration-directory*/Lev1/Logs**).

- The log files include a header that identifies the server's host machine, operating system, operating system release number, SAS release number, and server start-up command.

- The workspace server is not initially configured to create a rolling log file. If you need to create a rolling log file for this server for troubleshooting purposes, see "Create a Log File for Workspace Server Troubleshooting" on page 141.

SAS Management Console (**Log** tab)
For metadata servers, object spawners, OLAP servers, and stored process servers, the logging configuration files include an IOMServerAppender definition. This appender routes events to the **Log** tab in SAS Management Console and displays them using the same layout that is used for the rolling files.

For information about viewing logs in SAS Management Console, see "Use the Server Manager Tabs to Obtain Monitoring Information and to Adjust Logging Levels (If Needed)" on page 115.

CAUTION:
   Do not modify the message layout in the IOMServerAppender definitions that are provided in default logging configuration files. Modifying the layout will result in unpredictable behavior. However, you can adjust the number of messages that are captured by changing the value for the MAXENTRIES parameter.

Operating system facilities
For object spawners, OLAP servers, workspace servers, DATA step batch servers, and SAS/CONNECT servers, the logging configuration files include definitions of system facility appenders (WindowsEventAppender, UNIXFacilityAppender, or ZOSFacilityAppender). These appenders route events to the appropriate operating system facility.

The layout for these events includes the system date and time and the message text.

ARM logging facility
The logging configuration files for the following servers include definitions of ARMAppender, which collects performance-related events that can be processed by an ARM 4.0 server.

- Workspace servers and SAS/CONNECT servers. The logging configuration files for these servers include definitions of sLogAppender. This appender is configured to make performance-related events that are collected by ARMAppender available for display in SAS Data Integration Studio. You do not need to make any configuration changes to enable capture of these events.

> *CAUTION:*
>> **Do not modify sLogAppender or ARMAppender definitions that are provided in default logging configuration files. Modifying these definitions will result in unpredictable behavior. Because sLogAppender is a reserved class, you should not define new instances of this appender.**

- OLAP servers and DATA step batch servers. To enable the capture of ARM events for these servers, you must make configuration changes. See "Collecting ARM Log Information in SAS OLAP Servers" on page 144 and "Collecting ARM Log Information for SAS Data Integration Studio Jobs" on page 142.

For more information, see the *SAS Interface to Application Response Measurement (ARM): Reference*.

Modifying Your Server Logging Configurations

As a best practice, you should use the initial logging configuration files that are installed by the SAS Deployment Wizard. However, in some situations it might be necessary to make changes to the logging configuration. You can use the following methods to modify a server's logging configuration:

- You can adjust logging levels dynamically by using the Server Management features of SAS Management Console. This feature is useful when you want to change the configuration temporarily without stopping the server or spawner. See "Using SAS Management Console to Monitor SAS Servers" on page 112.

- You can permanently modify a server's logging configuration by changing its logconfig.xml file. Before modifying a configuration file, be sure to make a backup copy. Here are some examples of changes that you might want to make to a server's logging configuration:

    - configure RollingFileAppender to use a different log filename, to roll over log files more or less frequently, or to roll over log files based on file size rather than date

    - specify additional appenders (for example, you might want to define a separate rolling file to capture a particular category of messages)

    - use filters to limit the events that are written to an appender

    - configure a different message layout for an appender

    For additional options and detailed configuration instructions, see the *SAS Logging: Configuration and Programming Reference*.

- You can use the alternative logging configuration files that are provided for troubleshooting. These files are useful when SAS Technical Support requests a higher level of logging. For details, see the following topics:

    - "Enable More Detailed Logging for Server Troubleshooting"

    - "Create a Log File for Workspace Server Troubleshooting" on page 141

    - "Enable More Detailed Logging for SAS Object Spawner Troubleshooting" on page 142

- If you have installed the SAS Enterprise BI Audit and Performance Measurement package, you can use the alternative logging configuration files (called

logconfig.apm.xml) that are provided for use with the package. The package is available for download from http://support.sas.com/rnd/emi.

The following topics describe other logging changes that you might want to make:

CAUTION:
   Excessive logging can degrade performance. Therefore, you should not use the TRACE and DEBUG logging levels unless you are directed to do so by SAS Technical Support.

See Also

"Audit Messages for SAS Library Access" and "Audit Messages for Metadata-Bound Libraries" in the *SAS Logging: Configuration and Programming Reference*

Enable More Detailed Logging for Server Troubleshooting

To assist in troubleshooting, alternative logging configuration files are provided for some servers, including metadata servers, OLAP servers, pooled workspace servers, stored process servers, and workspace servers. These configuration files, which are named logconfig.trace.xml, provide high-level logging messages (for example, DEBUG and TRACE messages) which can be used by SAS Technical Support to help resolve server issues. The messages are written to the server's rolling log file.

Note: The workspace server is not initially configured to produce a log file. However, the alternative logging configuration file creates a rolling log file for this server for troubleshooting purposes. For details, see "Create a Log File for Workspace Server Troubleshooting" on page 141.

If SAS Technical Support requests that you enable detailed logging for server troubleshooting, follow these steps:

1. Modify the LOGCONFIGLOC= system option in the server's sasv9_usermods.cfg file to point to the path and filename of the alternative logging configuration file (logconfig.trace.xml), as in the following example:

   ```
   -logconfigloc "SAS-configuration-directory/Lev1/SASMeta/
   MetadataServer/logconfig.trace.xml"
   ```

   If there is an existing LOGCONFIGLOC= system option, comment it out and replace with the new option.

Each server's sasv9_usermods.cfg file is located in the directory where the server's start-up script is located. For example, the metadata server's sasv9_ usermods.cfg file is located in the following path:

SAS-configuration-directory/Lev1/SASMeta/MetadataServer

2. When troubleshooting is completed, revert the LOGCONFIGLOC= system option to the previous value.

Note: You can also adjust logging dynamically by using the Server Management features of SAS Management Console. See "Using SAS Management Console to Monitor SAS Servers" on page 112.

CAUTION:
 Do not use or modify the logconfig.trace.xml logging configuration files unless you are requested to do so by SAS Technical Support. Performance issues can result from the use of these files.

Create a Log File for Workspace Server Troubleshooting

Standard (non-pooled) workspace servers are not initially configured to produce log files. However, you might need to capture workspace server messages in a log file for troubleshooting purposes. In these situations, you can use the alternative logging configuration file (logconfig.trace.xml) that is provided in each workspace server's configuration directory.

The alternative configuration provides high-level logging messages (for example, DEBUG and TRACE messages) which can be used by SAS Technical Support to help resolve server issues. The messages are written to a rolling log file. The file is written to a Logs directory within the server's configuration directory.

CAUTION:
 Do not use or modify the logconfig.trace.xml logging configuration files unless you are requested to do so by SAS Technical Support. Performance issues can result from the use of these files.

Note: The default permissions to the path that is specified in the logconfig.trace.xml file (**/opt/sas/config/Lev1/SASApp/WorkspaceServer/Logs**) are sufficient. If they have been changed, you must ensure that Read, Write, and Execute permissions are set for the user that owns the directory and the group.

If SAS Technical Support requests that you enable detailed logging for workspace server troubleshooting, follow these steps:

1. Modify the LOGCONFIGLOC= system option in the workspace server's sasv9_usermods.cfg file to point to the path and filename of the alternative logging configuration file (logconfig.trace.xml), as in the following example:

-logconfigloc "SAS-configuration-directory/Lev1/SASApp/ WorkspaceServer/logconfig.trace.xml"

If there is an existing LOGCONFIGLOC= system option, comment it out and replace with the new option.

The workspace server sasv9_usermods.cfg file is located in the following path:

SAS-configuration-directory/Lev1/SASApp/WorkspaceServer

2. When troubleshooting is completed, revert the LOGCONFIGLOC= system option to the previous value.

Enable More Detailed Logging for SAS Object Spawner Troubleshooting

When resolving issues with the SAS object spawner, SAS Technical Support might request that you increase the logging level. To do so, you can use either of the following methods:

- To increase the logging level for multiple spawner sessions (if requested by SAS Technical Support):

  1. Open the spawner's logconfig.xml file, and locate the following lines:

     ```
     <logger name="App">
        <level value="Info">
     </logger>
     ```

  2. Change **<level value="Info">** to **<level value="Trace">**

  3. Restart the spawner.

  4. When troubleshooting is completed, open logconfig.xml again, change the logging level for the App logger back to **Info**, save the changes, and restart the spawner.

- To temporary increase the logging level (if requested by SAS Technical Support):

  1. Connect to the object spawner in SAS Management Console.

  2. On the **Loggers** tab, change the App logger to the logging level that was requested by SAS Technical Support. It is not necessary to restart the spawner.

  3. 'When troubleshooting is complete, return the logging level for the App logger to Information. It is not necessary to restart the spawner.

For details about using SAS Management Console to dynamically configure and view logs, see "Using SAS Management Console to Monitor SAS Servers" on page 112.

CAUTION:
   Excessive logging can degrade performance. Do not use the DEBUG or TRACE logging levels unless you are directed to do so by SAS Technical Support.

Collecting ARM Log Information for SAS Data Integration Studio Jobs

About Collecting ARM Log Information for SAS Data Integration Studio Jobs

When jobs are run interactively in SAS Data Integration Studio, ARM logging is used to collect performance-related events, which are displayed in the SAS Data Integration

Studio application. You do not need to change the initial server configuration to enable the capture and display of these events.

To collect performance-related events for SAS Data Integration Studio jobs that are run in batch mode, you must change the configuration of the SAS DATA step batch server that executes the jobs. The logging configuration files for SAS DATA step batch servers include specifications for collecting ARM log information and sending it to a log file. However, logging is not enabled by default for these servers. You must configure your server so that logging is enabled.

Note: If you have selected the **Collect Runtime Statistics** option for a SAS Data Integration Studio job, but ARM logging is not configured correctly for the DATA step batch server, then messages similar to the following will appear in the log when the job is run: WARNING: ARMINIT: Negative appid returned - check macro parameters, appid=-1(0xFFFFFFFF).

Configure DATA Step Batch Servers on Windows or UNIX to Collect ARM Log Information

To configure your DATA step batch server on Windows or UNIX so that ARM logging is enabled for SAS Data Integration Studio Jobs, follow these steps:

1. Log on to SAS Management Console as a user who has the appropriate credentials to create or update server metadata. See "Who Can Do What in the Server Manager Plug-in" on page 41.

2. Expand the **Server Manager** node.

3. Expand the node for the application server that executes your SAS Data Integration Studio jobs (for example, **SASApp**). Then expand the node for the logical DATA step batch server (for example, **SASApp - Logical DATA Step Batch Server**).

4. Right-click the node for the physical DATA step batch server (for example, **SASApp - SAS DATA Step Batch Server**), and select **Properties**.

5. Select the **Options** tab, and add the following expression to the contents of the **Command Line** field:

   ```
   -logconfigloc "file-specification"
   ```

   In *file-specification*, specify the fully qualified path to the logging configuration file. Here is an example command line for a Windows system with the logging option added:

   ```
   C:\SAS\Config\Lev1\SASApp\BatchServer\sasbatch.bat
   -logconfigloc "C:\SAS\Config\Lev1\SASApp\BatchServer\logconfig.xml"
   ```

   Here is an example command line for a UNIX system with the logging option added:

   ```
   /opt/SAS/Lev1/SASApp/BatchServer/sasbatch.sh
   -logconfigloc "/opt/SAS/Lev1/SASApp/BatchServer/logconfig.xml"
   ```

6. On the **Options** tab, click **Advanced Properties**. In the Advanced Properties dialog box, delete the contents of the **Rolling Log Options** field and replace them with a single underscore character (_).

7. Click **OK** to close the Advanced Properties dialog box, and click **OK** to close the **Options** tab.

Configure DATA Step Batch Servers on z/OS to Collect ARM Log Information

You can use the following procedure to configure your DATA step batch server on z/OS so that ARM logging is enabled for a SAS Data Integration Studio job:

1. Create a configuration file with the following contents:

```
**************************** Top of Data ************************
000001 logconfigloc=/<SAS-configuration-directory>/+
000002  Lev1/SASApp/BatchServer/logconfig.xml
000003 log=/<SAS-configuration-directory>/+
000004  Lev1/SASApp/BatchServer/Logs/<name-of-job>
**************************** Bottom of Data **********************
```

   Note:

   • Replace **<SAS-configuration-directory>** with the location that is specified in the CONFIG_DIR environment variable.

   • If your application server has a name other than SASApp, then replace **SASApp** with the appropriate characters (for example, **SASMain**).

   • Replace **<name-of-job>** with the name of the SAS Data Integration Studio job for which you want to collect ARM log information.

2. Save the configuration file in your SAS installation location.

3. In the JCL for each job that you want to collect ARM log information for, modify the //SASSTEP statement as follows so that it refers to the configuration file:

```
//SASSTEP  EXEC PROCENW0,CONFIG=<SAS-installation-
  location>.<configuration-filename>(DSBS)
```

   For example:

```
//SASSTEP  EXEC PROCENW0,CONFIG=SVS02.V920.KJB.CONFIG(DSBS)
```

Collecting ARM Log Information in SAS OLAP Servers

Overview

By default, SAS OLAP Servers collect log information in a SAS system log and an ARM log. The system log is displayed in the **Log** tab of the Server Manager when you select a SAS OLAP Server. The system log is also stored in a file that is refreshed daily. The ARM log is stored in a file that is created at server start. To change the default logging behavior, see "Initial Logging Configuration for SAS Servers" on page 135.

The ARM log receives no messages under normal operation, because the logger (Perf.ARM.OLAP_Server) is set to the WARN level.

During the testing and tuning of cubes, you change the logging level from WARN to INFO so that the ARM log begins to receive messages.

The Aggregation Tuning plug-in in SAS OLAP Cube Studio reads ARM logs to generate aggregations for common queries, as described in the *SAS OLAP Cube Studio: User's Guide*.

Be sure to truncate or delete large log files as necessary to maintain free space on disk.

CAUTION:
> **To ensure that you capture accurate log information, configure load-balancing clusters of SAS OLAP Servers so that each server writes to a separate ARM log file.**

For information about configuring ARM logs in load-balancing clusters, see the "Understanding Load Balancing" chapter of the *SAS Intelligence Platform: Application Server Administration Guide*.

Collect ARM Log Information without Restarting the SAS OLAP Server

Follow these steps to enable ARM logging without restarting the SAS OLAP Server:

Prerequisite
> Required OLAP server system option: ARMAGENT=LOG4SAS. This value is set by default when you installed the server.

1. Start SAS Management Console and select a connection profile that has Administer permission. Authenticate yourself as directed to connect to a metadata server.

2. Expand the **Server Manager** node, the SAS Application Server node (for example, **SASApp**), the node for the logical OLAP server, and the OLAP server instance.

3. Right-click the OLAP server instance and select **Connect**. Tabs appear in the right pane.

4. In the **Loggers** tab, right-click the **Perf.ARM.OLAP_Server** logger and select **Properties**.

5. In the Properties window, select the **Assigned** radio button.

6. In the **Assigned** field, click the down arrow and select **Information**.

7. Click **OK** to close the Properties window and begin logging.

8. Submit test queries to collect information in the ARM log file.

9. To disable logging, change the **Assigned** field back to the **Warning** level.

Audit Server Logging Level Changes That Originate in Client Applications

For security purposes, you might want to audit changes to server logging levels that are initiated by remote client applications. SAS Management Console is the only SAS application that enables users to change logging levels. The App.FilteredList.FilteredList.SetValue logger captures any logging level changes that are initiated by SAS Management Console or by custom applications.

To capture these events, add the following lines to the server's logging configuration file:

```
<logger name="App.FilteredList.FilteredList.SetValue">
   <level value="Trace">
```

```
</logger>
```

If you want, you can direct these events to a separate log file. To do so, define a new appender, and reference the appender in the logger definition. For details, see the *SAS Logging: Configuration and Programming Reference*.

You can also use SAS Management Console to temporarily adjust the threshold level for this logger. See "Using SAS Management Console to Monitor SAS Servers" on page 112 .

Collect Pre-assigned Library Events for Use in Troubleshooting

When resolving issues with library pre-assignment, SAS Technical Support might request that you change a server's logging configuration to collect pre-assignment events.

To capture these events, add the following lines to the server's logging configuration file:

```
<logger name="App.Program.Autoexec" additivity="false">
   <level value="Info">
</logger>
```

If you want, you can direct these events to a separate log file. To do so, define a new appender, and reference the appender in the logger definition, as in the following example:

```
<appender name="PreAssign" class="FileAppender">
   <param name="File" value="pgm-auto.log"/>
   <param name="ImmediateFlush" value="TRUE"/>
   <layout>
      <param name="ConversionPattern" value="%d %c - %m"/>
   </layout>
</appender>

<logger name="App.Program.Autoexec" additivity="false">
   <level value="info"/>
   <appender-ref ref="PreAssign"/>
</logger>
```

You can also use SAS Management Console to temporarily adjust the threshold level for the logger. See "Using SAS Management Console to Monitor SAS Servers" on page 112.

Audit Changes to Server Logging Configurations

For security purposes, you might want to audit changes that are made to a server's logging configuration file. Effective with SAS 9.4, the following loggers capture these changes:

• Audit.Logging.Configuration.Logger

• Audit.Logging.Configuration.Appender

- Audit.Logging.Configuration.Config

To capture these events, add the following lines to the logging configuration file that you want to audit:

```
<logger name="Audit.Logging.Configuration">
   <level value="Trace"/>
</logger>
```

If you want, you can direct these events to a separate log file. To do so, define a new appender, and reference the appender in the logger definition. For details, see the *SAS Logging: Configuration and Programming Reference*.

You can also use SAS Management Console to temporarily adjust the threshold level for these loggers. See .

Capture Metadata Server Management Events in the System Facility Appender

The App.Meta.Mgmt logger logs metadata server management activity such as the following:

- server operation actions such as pause, resume, and refresh

- creation and deletion of repositories

- modification of repository access modes

- repository backup and migration

You might want to capture these events in your operating system facility. To do so, add an instance of the appropriate system facility appender to the metadata server's logging configuration file, and direct App.Meta.Mgmt events to that appender.

For Windows systems, add the following lines to the logging configuration file:

```
<appender name="eventLog" class="WindowsEventAppender">
   <param name="Threshold" value="Warn"/>
   <layout>
      <param name="ConversionPattern" value="%d{yyyyMMdd:HH.mm.ss.SS}:
%m"/>
   </layout>
</appender>
<logger name="App.Meta.Mgmt">
   <level value="Info"/>
   <appender-ref ref="eventLog"/>
</logger>
```

On UNIX systems, replace **WindowsEventAppender** with **UNXFacilityAppender**. On z/OS systems, replace **WindowsEventAppender** with **ZOSFacilityAppender**.

Capture XML Information in the SAS Metadata Server Log

About Capturing XML Information

To assist in troubleshooting, SAS Technical Support might request that you capture generated XML entries in the metadata server log.

When clients communicate with the SAS Metadata Server to create, update, delete, or query a metadata object, information about the metadata object is represented in XML. The metadata server also uses XML entries to return information about metadata objects to clients. For example, to add a metadata object to a repository, clients use the AddMetadata method of the SAS Open Metadata Interface. One of the parameters for this method is an XML description of the object to be added. If a client requests that a table object be added, then the request includes an XML description of the table, as shown in this example:

```
<PhysicalTable Name="NECust" Desc="All customers in the northeast region"/>
```

The metadata server returns an XML string that includes the input description as well as the object ID of the new metadata object.

These XML entries can be captured in the metadata server log (or in a separate log file) and used for troubleshooting. To capture the entries, use the one of the following procedures (or a combination of these procedures) as appropriate:

- "Capture XML Information in the Metadata Server Log for the Current Server Session Only"

- "Capture XML Information in the Metadata Server Log across Multiple Server Sessions"

Note: For more information about the call interfaces, see *SAS Language Interfaces to Metadata*.

Capture XML Information in the Metadata Server Log for the Current Server Session Only

If you want to capture only XML information during the current metadata server session, follow these steps:

1. Open the Server Manager in SAS Management Console and connect to the metadata server.

2. On the **Options** tab, change the settings for the **IOM.JnlStrMax** and **IOM.JnlLineMax** properties to a higher value (for example, 1000000). The parameters are explained in the following table:

Parameter	Explanation
JNLSTRMAX	Specifies the maximum journal string length. The default is 500 characters. If this parameter is omitted, a maximum of 500 characters of the XML string is logged, and the following message is appended to the XML: ***truncated to 500 characters (jnlStrMax)**.
JNLLINEMAX	Specifies the maximum journal line length. The default is 1000 characters. If this parameter is omitted, a maximum of 1000 characters of the XML string is logged, and the following message is appended to the XML: ***truncated to 1000 characters (jnlLineMax)**.

3. On the **Loggers** tab, increase the logging level for the App.OMI logger to TRACE.

   The XML information will be captured in the metadata server log for the duration of the current server session, or until you return the App.OMI logger to its default level (Inherited). To avoid adverse effects on performance, you should reset the logger when you are finished capturing XML information.

Capture XML Information in the Metadata Server Log across Multiple Server Sessions

If you want to capture XML information across multiple metadata server sessions, follow these steps:

1. Update the object server parameters in the metadata server's configuration file as follows:

   a. Open the sasv9.cfg and sasv9_usermods.cfg files in a text editor. The files are located in **SAS-configuration-directory/Lev1/SASMeta/ MetadataServer**.

   b. In sasv9.cfg, find the line that specifies object server parameters. The line begins with **-objectserverparms** (or **objectserverparms** for z/OS). Copy this line to sasv9_usermods.cfg.

   c. In sasv9_usermods.cfg, add JNLSTRMAX and JNLLINEMAX to the existing object server parameters. Enclose the entire list of parameters in quotation marks, as shown in the following example:

   ```
   -objectserverparms "jnlStrMax=1000000 jnlLineMax=1000000 cel=credentials"
   ```

   CAUTION:
   > **The typical configuration file contains more object server parameters than are shown in this example. Make sure not to delete or overwrite the existing parameters.**

   You can specify different values for JNLSTRMAX and JNLLINEMAX than those that are shown in the example.

2. Open the server's logconfig.xml file (located in **SAS-configuration-directory/Lev1/SASMeta/MetadataServer**), and add the following lines:

   ```
   <logger name="App.OMI">
       <level value="trace"/>
       <appender-ref ref="IOMCALLTRACE"/>
   ```

```
</logger>
```

If you want to capture the XML information n a separate log file (instead of in the metadata server log), insert the following lines before the App.OMI logger element:

```
<appender class="FileAppender" name="IOMCALLTRACE">
   <param name="File" value="omasrv_iomcalltrace.log"/>
   <param name="ImmediateFlush" value="true"/>
   <param name="Append" value="false"/>
   <layout>
     <param name="ConversionPattern" value="%d %-5p [%t] %c
           (%F:%L) - %m"/>
   </layout>
</appender>
```

If you want, you can specify different values (for example, a different log filename) for the FileAppender parameters. For details, see the FileAppender topic in the *SAS Logging: Configuration and Programming Reference*.

3. Restart the metadata server.

To avoid adverse effects on performance, you should remove the App.OMI logger from logconfig.xml and restart the server when you are finished capturing XML information.

Examples of XML Log Entries

The format of the XML that is written to the log depends on whether the request was made using the standard SAS Open Metadata Interface or using the DoRequest method. Here is an example of a log entry for the receipt of a GetMetadataObjects request:

```
2010-11-06T15:37:55,815 TRACE [00000033] App.OMI.OMI.DoRequest
(CIOMomici.c:6239) - IOM CALL {compRef:5e274b0}->OMI::DoRequest():
 inString=<GetMetadataObjects><Reposid>A0000001.A0000001</Reposid>
<Type>RepositoryBase</Type><Objects/><ns>REPOS</ns>
<Flags>264</Flags><options/></GetMetadataObjects>
```

Here is an example of a log entry for a response to a client request:

```
2010-11-06T15:37:55,815 TRACE [00000033] App.OMI.OMI.DoRequest
(CIOMomici.c:6504) - IOM RETURN 0={compRef:5e274b0}->OMI::DoRequest():
 outString=<GetMetadataObjects><Reposid>A0000001.A0000001</Reposid>
<Type>RepositoryBase</Type><Objects><RepositoryBase Id="A0000001.A510HYM9"
Name="Foundation" Access="0" Desc="" Engine="BASE"
MetadataCreated="26Sep2010:16:12:48" MetadataUpdated="02Nov2010:18:10:37"
Options=""
Path="C:\SAS\Config\Lev1\SASMeta\MetadataServer\MetadataRepositories\Foundation"
RepositoryFormat="11" RepositoryType="FOUNDATION"/><RepositoryBase
Id="A0000001.A5PJCXVL" Name="Custom1" Access="0" Desc="" Engine="BASE"
MetadataCreated="26Sep2010:16:14:28" MetadataUpdated="02Nov2010:18:10:37"
Options=""
Path="C:\SAS\Config\Lev1\SASMeta\MetadataServer\MetadataRepositories\Custom1"
RepositoryFormat="11" RepositoryType="CUSTOM"/><RepositoryBase
Id="A0000001.A5FIU3O4" Name="Project1" Access="0" Desc="" Engine="BASE"
MetadataCreated="26Sep2010:16:15:53" MetadataUpdated="02Nov2010:18:10:37"
Options=""
Path="C:\SAS\Config\Lev1\SASMeta\MetadataServer\MetadataRepositories\Project1"
RepositoryFormat="11" RepositoryType="PROJECT"/></Objects><ns>REPOS</ns>
<Flags>264</Flags><options/></GetMetadataObjects> retval=0
```

Part 4

Backing Up and Restoring Your SAS Content

Chapter 11
About Backing Up and Restoring Your SAS Content

Best Practices for Backup and Restore

In order to guarantee the integrity of your SAS environment, you must implement the following as part of your backup and restore strategy:

1. Instigate a regular file system backup of your entire SAS deployment by using SAS backup facilities along with operating system commands or third-party tools.

    *   You can use the SAS Metadata Server backup facility or operating system commands to back up the SAS Metadata Server. For more information, see "About the Metadata Server Backup Facility" on page 155.

        Note: If you choose to use operating system commands, then you must be sure to pause the metadata server to an Offline state before performing the backup. For more information, see "Using Operating System Commands to Back Up the Metadata Server" on page 156.

    *   You must use operating system commands or third-party tools to back up the following:

        *   databases from third-party vendors (when the SAS Web Infrastructure Platform Data Server is not used)

        *   SAS installation files

        *   complete contents of the **SAS-configuration-directory/Levn** directory

        *   SAS data files that contain source data for SAS applications or solutions

        *   reference locations for symbolic links

        *   batch server components and the SAS Grid Server component.

2. Synchronize the backups of the SAS Metadata Server, SAS Content Server, SAS Web Infrastructure Platform Data Server (or databases from third-party vendors), and Physical File Backups. If your database is managed by a database administrator rather than you as the SAS administrator, then you must coordinate with the database administrator to synchronize database backups with backups of the other components. For more information, see "Synchronize Metadata Backups with SAS Content Server, SAS Web Infrastructure Platform Data Server, and Physical File Backups" on page 165.

3. The Deployment Backup Recovery tool, new with SAS 9.4, backs up only a subset of the components in your deployment and does not support the backup and recovery of third-party databases. It is not intended to replace a separate file system backup of the entire deployment.

    Using the Deployment Backup and Recovery Tool as part of a larger backup and recovery strategy has the following benefits:

    *   Backups are automatically scheduled and are retained for a period of 30 days.

    *   SAS Content is automatically detected across multiple tiers and machines by connecting with the SAS Deployment Agent on each machine.

    *   Backups of included components are synchronized.

    *   You can administer the tool with batch commands or with the SAS Backup Manager. For more information, see "Overview of Backup and Recovery

About Backups and Restores

To ensure the integrity of the content that is created and managed by the SAS Intelligence Platform, you should establish a formal, regularly scheduled backup process. It is important to back up all of the following content items at the same point in time so that related information will be synchronized if a restore becomes necessary:

- all of your metadata repositories, the repository manager, and the files in the metadata server configuration directory. The SAS Metadata Server includes a server-based facility that creates correct backups of these files.

- the SAS Content Server, the databases that are managed by the SAS Web Infrastructure Platform Data server, server configuration information, and the physical content that is stored in files and databases. To back up these resources, you can use the Deployment Backup and Recovery tool, which is new with SAS 9.4. Alternatively, you can use operating system commands or third-party tools to back up these resources.

In addition to performing regular full backups, in some situations it might be appropriate to back up specific objects or folders in the metadata folders (SAS Folders) tree. In these situations, you can use the promotion tools, which include the Export SAS Package wizard, the Import SAS Package wizard, and the batch export and import tools. For details, see Chapter 20, "Promotion Tools Overview," on page 265.

Before you back up your SAS content, read the following information:

- "About the Metadata Server Backup Facility" on page 155
- "Using Operating System Commands to Back Up the Metadata Server" on page 156
- "About Using the Export SAS Package Wizard to Back Up Specific SAS Folders" on page 157
- Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153
- "About the Deployment Backup and Recovery Tool" on page 157
- "Best Practices for Backing Up Your SAS Content" on page 159
- "Best Practices for Restoring Your SAS Content" on page 167

> ***CAUTION:***
> **If you choose to use operating system commands to back up your metadata repositories and the repository manager, instead of using the metadata server's backup facility, then you must be sure to pause the metadata server to an Offline state before performing the backup. If the metadata server is Online or is paused to an Administration state, then the backup files will not be usable.**

About the Metadata Server Backup Facility

The SAS Metadata Server includes a server-based facility that performs metadata server backups automatically, on a scheduled basis, without the need for administrator

intervention. This facility can also be used to perform ad hoc backups and to perform roll-forward recovery.

The backup facility backs up all registered metadata repositories, the repository manager, and the metadata server's configuration directory. By default, metadata server backups are scheduled to run at 1:00 a.m. every day except Sunday. The Deployment Backup and Recovery tool, if it is configured, backs up the metadata server (along with other resources) each Sunday at 1:00 a.m. by default. Administrators can change this schedule as needed. To avoid conflicts, stand-alone metadata server backups and deployment backups should not be scheduled in the same backup window.

Metadata server backups can also be launched from SAS Management Console, from the operating system command line, from SAS, or through third-party scheduling software.

The metadata server backup facility provides the following advantages:

- The backup facility executes in a separate thread while the metadata server is running. Therefore, the server does not need to be paused during backups unless certain options are selected. When repositories are restored in the event of a metadata server failure, transactions recorded in the journal file can be used to perform roll-forward recovery to a specified point in time after the last backup was taken.

- The metadata server runs its own backup and recovery processes and does not rely on any other servers. Because the metadata server has permission to access all of the needed files, there is no need to set up a host account with special operating system rights and permissions.

- The backup facility automatically includes all of the metadata server's data sets, including the repository manager, all of your registered metadata repositories, and the individual files that are stored in the metadata server configuration directory (*SAS-configuration-directory*/**Lev1/SASMeta/MetadataServer**).

- SAS Management Console displays a detailed history of backups that were performed using the backup facility. Status icons indicate which backups are active and which backups have been deleted based on the retention policy. When you view the properties of a backup, the backup files are validated, and the status icon is updated to indicate whether the backup is usable.

- When repositories are restored in the event of a metadata server failure, transactions recorded in the journal file can be used to perform roll-forward recovery to a specified point in time after the last backup was taken.

See Also

Using Operating System Commands to Back Up the Metadata Server

Instead of using the metadata server backup facility, you can choose to use operating system commands to back up your metadata repositories and the repository manager. If you use this method, then you must be sure to pause the metadata server to an Offline state before performing the backup. If the metadata server is Online or is paused to an Administration state, then the backup files will not be usable.

Note: External systems such as Windows Services Manager report the Offline state as
`paused`.

You can use PROC METAOPERATE to pause the server to an Offline state before the
backup is taken and to resume the server to an Online state when the backup is complete.
For details, see the following:

- "Example of a PROC METAOPERATE Program That Pauses the Metadata Server to an Offline State" on page 166

- "Example of a PROC METAOPERATE Program That Resumes the Metadata Server" on page 166

You can also pause and resume the server by using SAS Management Console or a
server management script. See Chapter 6, "Operating Your Servers," on page 61. Be sure
to pause the server to an Offline state (rather than the Administration state).

About Using the Export SAS Package Wizard to Back Up Specific SAS Folders

In addition to performing regular full backups, it might be appropriate in certain
situations to make backups of specific SAS folders that are used to organize user-created
SAS content. To perform selective backups and restores of objects in these folders (as
well as physical files associated with these objects), you can use the following tools:

- the Export SAS Package and Import SAS Package wizards, which are available in
  the **Folders** tab of SAS Management Console. The same wizards are also available
  in SAS Data Integration Studio and SAS OLAP Cube Studio.

- the batch export tool and the batch import tool, which enable you to perform exports
  and imports on a scheduled or repeatable basis. These tools provide most of the same
  capabilities as the Export SAS Package and Import SAS Package wizards.

Here are some examples of situations in which you might want to back up individual
folders:

- If you are creating new content (for example, new information maps and new
  reports) in a development environment, you might want to periodically back up the
  folders and associated physical files that contain this content.

- If you are about to import new content (for example, new information maps, stored
  processes, and reports) into production folders, you might want to first back up the
  current contents of the production folders and their associated physical files.

See Also

For details about using the Export SAS Package and Import SAS Package wizards and
the batch import and export tools, see Chapter 20, "Promotion Tools Overview," on page
265.

About the Deployment Backup and Recovery Tool

The Deployment Backup and Recovery tool, new with SAS 9.4, provides an integrated
method for backing up and recovering your SAS content across multiple tiers and

machines. The tool is installed on the middle tier as part of the SAS Web Infrastructure Platform. It connects with the SAS Deployment Agent on each middle-tier and server-tier host machine and backs up the following components:

- the metadata server, including all registered metadata repositories, the repository manager, and the server's configuration directory.

- the contents of the **Data** directories, **SASEnvironment** directories, and server configuration directories for each server on the SAS server tier. (If symbolic links in these directories point to other locations, the referenced locations are not backed up.)

- the SAS Content Server repository.

- the databases that are managed by the SAS Web Infrastructure Platform Data Server. By default, all of the databases are backed up. You can modify the backup configuration so that only selected databases are backed up.

- additional directories under **SAS-configuration-directory/Levn** as specified by the administrator.

By default, the Deployment Backup and Recovery tool backs up these items automatically each Sunday at 1:00 a.m. Backup files are retained for a period of 30 days. Batch commands are available on each host machine to use in administering backups and recoveries.

Note: If you are using a third-party vendor database (instead of the SAS Web Infrastructure Platform Data Server) for the SharedServices database, the Deployment Backup and Recovery Tool does not back up or recover the third-party vendor database. You must manually back up and restore the data. For more information, see "Backup and Restore of Third-Party Databases" on page 158.

See Also

Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387

Backup and Restore of Third-Party Databases

The Deployment Backup and Recovery Tool does not support third-party databases. Therefore, if your deployment uses a third-party vendor database for the SharedServices database rather than the default database that is managed by the SAS Web Infrastructure Platform Data Server, you must take alternative steps to back up and restore that third-party database.

To ensure that all SAS Content Server components are covered, be sure to include the SAS Content Server tables in the SharedServices database in your backup. You can identify the SAS Content Server tables by searching for tables in the database that are prefixed with "sas_scs2_". You must also back up the following additional items for the SAS Content Server:

- the repository.xml file in **SAS-configuration-directory/Levn/AppData/ SASContentServer/Repository**

- the File Data Store, if you have configured it. This is an append-only store, and therefore it supports incremental backups.

About SAS Backup Manager

SAS 9.4M3 includes SAS Backup Manager, an easy-to-use interface for the Deployment Backup and Recovery tool. You can use SAS Backup Manager for the following tasks:

- view backup and recovery history
- run an immediate (ad hoc) backup
- view the backup configuration
- modify the backup configuration (except backup filters and custom directories)
- view information about backup and recovery sources
- view and modify the backup schedule

In previous SAS 9.4 releases, these functions were available only through batch commands.

SAS Backup Manager can be accessed from the **Administration** tab of SAS Environment Manager. For more information, see the *SAS Environment Manager 2.5 Administration: User's Guide*, *SAS Environment Manager 2.6 Administration: Fundamentals*, and the Help for SAS Backup Manager.

Best Practices for Backing Up Your SAS Content

Summary of Best Practices for Backing Up Your SAS Content

Here are the recommended best practices for backing up your SAS content:

- Always use the metadata server backup facility to back up the repository manager and the metadata repositories..
- Perform regularly scheduled full backups.
- Perform backups before and after major changes.
- Specify a reliable backup destination that is included in daily system backups.
- Occasionally use the Reorganize Repositories option to reclaim unused disk space.
- Back up the SAS Content Server.
- Back up the databases that are managed by the SAS Web Infrastructure Platform Data Server.
- Back up the third-party vendor databases that are managed by the SAS Web Infrastructure Platform Data Server.
- Back up data that is stored in file systems.
- Synchronize metadata backups with SAS Content Server, SAS Web Infrastructure Platform Data Server, and physical file backups.

The following topics describe these best practices.

Always Use the Metadata Server Backup Facility to Back Up the Repository Manager and the Metadata Repositories

The processes that you use to make regular system backups of your file system cannot directly access the metadata repositories, because the repository data sets are normally locked. Therefore, the recommended method for backing up your metadata repositories, the repository manager, and the metadata server's configuration directory is to use the metadata server backup facility. By default, the backup facility runs scheduled backups at 1:00 a.m. every day except Sunday. Administrators can change this schedule as needed. Your daily system backup process can then copy the files from the metadata server's backup destination.

The Deployment Backup and Recovery tool also uses the metadata server backup facility for the metadata server backups that it performs. This tool, if it is configured, is scheduled to run each Sunday at 1:00 a.m.

The advantages of the metadata server backup facility are described in "About the Metadata Server Backup Facility" on page 155.

CAUTION:
> **If you choose to use operating system commands to back up your metadata repositories and the repository manager, instead of using the metadata server's backup facility, then you must be sure to pause the metadata server before performing the backup. Otherwise, the backup files will not be usable.**

Perform Regularly Scheduled Full Backups

By default, the metadata server backup facility takes daily backups of your metadata repositories, the repository manager, and the files in the metadata server's configuration directory. To safeguard the integrity of your metadata, it is important to maintain this backup schedule. Daily use of the metadata server backup facility will enable you to recover the server to a particular point in time in the event of a server failure, while correctly maintaining associations between metadata.

To maintain system integrity, you must also back up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical files that are associated with your metadata. See the following topics:

- "Back Up the SAS Content Server" on page 162
- "Back Up the SAS Web Infrastructure Platform Data Server Databases" on page 164
- "Back Up Data That Is Stored in File Systems" on page 164
- "Synchronize Metadata Backups with SAS Content Server, SAS Web Infrastructure Platform Data Server, and Physical File Backups" on page 165

Perform Backups before and after Major Changes

In addition to your regular full backup schedule, you should also perform a full backup before and after making major changes to your system. For example, you should perform a backup on the following occasions:

- after you complete your initial installation and configuration
- before and after you update your SAS software to a new release

Note: In multiple-machine deployments, SAS recommends that you create backups at key points during the update process. For instructions, see "Create Restore Points for Multiple Machine Deployments" in *SAS Guide to Software Updates and Product Changes*.

- before and after you configure a new SAS product or remove the configuration of an existing product

- before and after you configure a new tier in your SAS deployment

- after you add a new metadata repository

- before you delete a metadata repository

The best way to perform an ad hoc (unscheduled) backup of the metadata server is to use the metadata server backup facility. You can launch this facility from SAS Management Console, from the operating system command line, from a SAS session, or through a third-party scheduling system.

In unscheduled backup situations that involve associated content and files (for example, the addition of new reports or stored processes), be sure to include the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical files in your backup. If the backup does not involve a large amount of data, then the Import and Export wizards can be useful for this type of backup. See "About Using the Export SAS Package Wizard to Back Up Specific SAS Folders" on page 157.

In unscheduled backup situations where the metadata has changed but no physical files are involved, it is sufficient to back up just the metadata server. For example, no physical files are affected when you make changes to server definitions, user and group definitions, and access control templates. If you choose not to back up the physical files, you should make sure that no user activity that affects physical files has taken place since the last backup was taken. For example, make sure that no stored processes, jobs, or reports have been created or modified since the last backup.

Specify a Reliable Backup Destination That Is Included in Daily System Backups

By default, the metadata server backup facility writes backup files to the **Backups** subdirectory of the metadata server configuration directory. Within the backup location, each set of backup files (along with the associated journal file) is stored in a directory that is named based on the date and time that the backup is started.

As a best practice, you should modify your backup configuration to specify a storage device other than the device that is used to store the metadata repositories and the server configuration files. Specifying a separate device ensures that backup files and their associated journal files are available in the event of a disk failure.

Make sure that the **Backups** directory (or the backup destination that you specify) is included in your regular system backups.

See Also
"Modifying the Metadata Server Backup Configuration (Backup Location and Retention Policy)" on page 175

Occasionally Use the Reorganize Repositories Option to Reclaim Unused Disk Space

When metadata is deleted from a metadata repository, the record is removed from both memory and disk. However, the disk space that had been allocated for the record remains in the data set after the record is removed.

When you use the metadata server backup facility, you can use the Reorganize Repositories option to reclaim unused disk space that is left from previously deleted metadata objects. The default backup schedule specifies a weekly reorganization. It is not necessary to reorganize the repositories more frequently than once a week, except in extraordinary situations such as the deletion of a large amount of metadata.

The repository reorganization process affects disk space only. It does not affect the memory usage of the metadata server.

See Also
"About the Reorganize Repositories Backup Option" on page 173

Back Up the SAS Content Server

Overview

If your deployment includes SAS web applications or the SAS Web Infrastructure Platform, then it is important to back up the SAS Content Server when you back up the metadata server. The SAS Content Server contains content that is associated with metadata objects that are contained in the repositories, including the following:

• content for the SAS Information Delivery Portal

• report definition files

• other supporting files for reports, including PDF files and image files

• content for SAS solutions

Metadata cannot be used without the associated content that is stored on this server. If you need to recover from a failure, the metadata that you restore must be synchronized correctly with the content server.

You can use the Deployment Backup and Recovery tool to back up the SAS Content Server. Alternatively, if you are storing SAS Content Server content in the file system, you can back it up as follows:

1. As a best practice, stop either the SAS Web Application Server or the SAS Content Server application before making the backup.

2. Use operating system commands or third-party tools to copy all of the files and subdirectories from the following path:

   ***SAS-configuration-directory*/Lev1/AppData/SASContentServer/ Repository**

If you need to back up just a subset of the SAS Content Server, or if you want to create a backup for input to a system other than the SAS Content Server, see "WebDAVDump and WebDAVRestore Utilities" on page 163.

Note: If you are using a third-party vendor database (instead of the SAS Web Infrastructure Platform Data Server) for SAS Content Server, the Deployment

Backup and Recovery Tool does not back up or recover the third-party vendor database. For more information, see "Backup and Restore of Third-Party Databases" on page 158.

WebDAVDump and WebDAVRestore Utilities

If you need to back up just a subset of the SAS Content Server, or if you want to create a backup for input to a system other than the SAS Content Server, you can use the WebDAVDump and WebDAVRestore utilities.

* WebDAVDump

  WebDAVDump will dump the contents of a WebDAV server while saving directories, content and properties. It creates one ZIP file that contains a file named WFSDump.xml that describes the content, and one file for each content resource found in the server. Run the utility by specifying the URL path to the SAS Content Server content to dump, the sasadm@saspw user and password, and the name and directory path of the ZIP file you wish to create.

  1. Navigate to the directory that contains the utility.

  Table 11.1 *WebDAVDump Directory Path*

Windows	`\`*`SAS-configuration-directory`*`\Lev1\Web\Utilities`
Unix	`/`*`SAS-configuration-directory`* `/Lev1/Web/Utilities`

  2. Run the command.

     * **Windows:**

       `WebDAVDump.bat http://`*`server`*`:`*`port`*`/SASContentServer/repository/default/sasfolders/`*`folder user password zip-file-path`*

     Note: If you are dumping a directory that has spaces in its name, put quotes around the whole path. For example:

     `WebDAVDump.bat "http://myserver:8080/SASContentServer/repository/default/sasfolders/my folder name" sasadm@saspw mypassword C:\mydump.zip`

     * **Unix:**

     `WebDAVDump.sh http://`*`server`*`:`*`port`*`/SASContentServer/repository/default/sasfolders/`*`folder user password zip-file-path`*

     Here is an example from a Unix environment:

     `WebDAVDump.sh http://myserver:8080/SASContentServer/repository/default/sasfolders/myfolder sasadm@saspw mypassword /usr/mydump.zip`

* WebDAVRestore

  WebDAVRestore will restore the contents of the dump ZIP file created by WebDAVDump to the WebDAV server. Run the utility by specifying the URL path to the SAS Content Server, the sasadm@saspw user and password, and the name and directory path of the ZIP file to restore.

  1. Navigate to the directory that contains the utility.

Table 11.2 *WebDAVRestore Directory Path*

Windows	`\`***SAS-configuration-directory*** `\Lev1\Web\Utilities`
Unix	`/`***SAS-configuration-directory*** `/ Lev1/Web/Utilities`

2. Run the command.

  • **Windows:**

  `WebDAVRestore.bat http://`*server*:*port*`/SASContentServer/repository/default /sasfolders/`*folder user password zip-file-path*

  Note: If you are dumping a directory that has spaces in its name, put quotes around the whole path. For example:

  `WebDAVRestore.bat "http://myserver:8080/SASContentServer/repository/default /sasfolders/my folder name" sasadm@saspw mypassword C:\mydump.zip`

  • **Unix:**

  `WebDAVRestore.sh http://`*server*:*port*`/SASContentServer/repository/default /sasfolders/`*folder user password zip-file-path*

  Here is an example from a Unix environment:

  `WebDAVRestore.sh http://myserver:8080/SASContentServer/repository/default /sasfolders/myfolder sasadm@saspw mypassword /usr/mydump.zip`

Back Up the SAS Web Infrastructure Platform Data Server Databases

If your deployment includes a SAS Web Infrastructure Platform Data Server, then it is important to back up the databases that are managed by the server. You can use the Deployment Backup and Recovery tool to back up all of these databases automatically.

Back Up the Third-Party Vendor Databases

Instead of using the default database that is managed by SAS Web Infrastructure Platform Data Server, your deployment might include third-party vendor databases to meet specific business requirements. In this case, you must manually back up and restore that third-party vendor database. The Deployment Backup and Recovery Tool cannot be used to back up a third-party vendor database. For more information, see "Backup and Restore of Third-Party Databases" on page 158.

Back Up Data That Is Stored in File Systems

When you back up the metadata server, it is important to also back up the physical data that is associated with the metadata objects that are contained in the repositories. The metadata cannot be used without the associated data. If you need to recover from a failure, the metadata that you restore must be synchronized correctly with the associated data.

You must use operating system commands or third-party tools to back up the physical data, which might include the following:

- portal content that is not stored in the SAS Content Server, including custom portlets, custom themes, and applications

- report definition files and other supporting files for reports (for example, PDF files and image files) that are stored in your file system, if you are not storing these items on the SAS Content Server

- source files for stored processes

- packages that have been published to a channel's persistent store (archive) location

- stored process results that are stored in permanent packages

- data sources, including the following:

  - SAS tables

  - OLAP data

  - SAS Scalable Performance Data Server data

  - RDBMS data sets

- custom code, processes, and postprocesses that are associated with SAS Data Integration Studio jobs

- XML files and XML maps that are contained in libraries

- source files for SAS Data Integration Studio jobs that have been scheduled or deployed for scheduling

Synchronize Metadata Backups with SAS Content Server, SAS Web Infrastructure Platform Data Server, and Physical File Backups

Suggested Approach for Synchronizing Metadata Backups with Physical Backups

To maintain system integrity in the event that a restore is needed, your backups of the SAS Content Server, SAS Web Infrastructure Platform Data Server, and physical data must be synchronized correctly with your metadata backups. To ensure that the backups are synchronized, you can use one of the following approaches:

- Back up the metadata server, the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical files concurrently (that is, in the same backup window). One way to do this is to use the Deployment Backup and Recovery tool.

- Back up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical files immediately after the metadata server is backed up, and do not allow clients to update metadata while you are performing these backups. If you are running the backup on a batch basis (for example, as part of a daily schedule), then you can do the following to implement this approach:

  1. Write a program that uses PROC METAOPERATE to pause the metadata server to an Offline state. See "Example of a PROC METAOPERATE Program That Pauses the Metadata Server to an Offline State" on page 166.

     You can use this program to pause the metadata server while you back up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and associated physical data. If you use operating system commands to back up the metadata server, then you can use this program to pause the server before running the backup.

2. Write another program that resumes the metadata server to an Online state. See "Example of a PROC METAOPERATE Program That Resumes the Metadata Server" on page 166. You can use this program after using operating system commands to back up the metadata server, or you can use it after backing up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and associated physical data.

• If you are running an ad hoc (unscheduled) backup and you need to also back up associated data, then you can do the following to prevent clients from updating metadata while you are backing up the associated data:

1. Use the metadata backup facility to back up the metadata server. Then immediately use SAS Management Console to pause the metadata server. As an alternative, you can use SAS Management Console to temporarily change the registered access mode of the repositories to ReadOnly.

2. Back up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical data.

3. When you are finished backing up the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical data, use SAS Management Console to resume the metadata server (or to return the registered access mode of the repositories to Online).

Example of a PROC METAOPERATE Program That Pauses the Metadata Server to an Offline State

Here is an example program that uses PROC METAOPERATE to pause the metadata server to an Offline state.

If you use operating system commands to back up the metadata server, you can use this program to pause the server before running the backup.

If you use the metadata server backup facility, it is not necessary to pause the server while you back it up. However, you can use this program to pause the metadata server while you back up the SAS Content Server, SAS Web Infrastructure Platform Data Server, and associated physical data.

```
proc metaoperate
        server="host-name"
        port=8561
        userid="user-ID"
        password="encoded-password"
        protocol=bridge
        action=pause
        noautopause;
run;
```

The user that is specified in *user-ID* must be assigned to the **Metadata Server: Operation** role.

Example of a PROC METAOPERATE Program That Resumes the Metadata Server

Here is an example program that uses PROC METAOPERATE to resume the metadata server to an Online state. You can use this program after using operating system commands to back up the metadata server, or you can use it after backing up the SAS Content Server, SAS Web Infrastructure Platform Data Server, and associated physical data.

```
proc metaoperate
```

```
                server="host-name"
                port=8561
                userid="user-ID"
                password="encoded-password"
                protocol=bridge
                action=resume
                noautopause;
run;
```

The user that is specified in *user-ID* must be assigned to the **Metadata Server: Operation** role.

For detailed information about PROC METAOPERATE, see *SAS Language Interfaces to Metadata*.

Best Practices for Restoring Your SAS Content

Perform Full Restores

Like full backups, full restores are the best practice to follow to ensure that the metadata server returns to a consistent state and that metadata associations are correctly maintained. Full restores include the following:

• the repository manager, all of your metadata repositories, and the metadata server's configuration files (including omaconfig.xml, adminUsers.txt, trustedUsers.txt, and logconfig.xml).

If you use the metadata server backup facility to create your backups, then you can use the metadata server recovery facility in the event of a server failure. The recovery facility automatically restores the repository manager and all of the repositories that are included in the backup. You can also choose to restore the contents of the metadata server's configuration directory and to perform roll-forward recovery to a specified point in time after the last backup was taken.

You can also use the Deployment Backup and Recovery tool to restore the metadata server from a backup that was performed by that tool.

• the SAS Content Server, the SAS Web Infrastructure Platform Data Server, and the physical files that are associated with the metadata. You can use the Deployment Backup and Recovery tool to restore these resources along with the metadata server.

Note: You cannot use the metadata server recovery facility on a metadata server that has been started with clustering. To recover a clustered metadata server, see "Using the -recover Option" on page 224.

Note: If you are using a third-party vendor database instead of the default database that is managed by SAS Web Infrastructure Platform Data Server, the Deployment Backup and Recovery Tool does not back up or recover the third-party vendor database. You must manually back up and restore the data. For more information, see "Backup and Restore of Third-Party Databases" on page 158.

Restoring an Unresponsive Metadata Server

The metadata server recovery facility can restore repositories only on a running metadata server. If your metadata server is not responding to client requests, then you must use other methods to restore the metadata repositories and the repository manager. For

instructions, see .

Chapter 12

Backing Up and Recovering the SAS Metadata Server

About the SAS Metadata Server Backup and Recovery Facility

Overview of the SAS Metadata Server Backup and Recovery Facility

The SAS Metadata Server includes a server-based facility that performs metadata server backups automatically on a scheduled basis. This facility can also be used to perform ad hoc backups and to perform roll-forward recovery. The backup and recovery facility provides the following features:

• Backups are taken without the need for administrator intervention, as follows:

- By default, metadata server backups are scheduled to run at 1:00 a.m. server local time every day except Sunday. The Deployment Backup and Recovery tool, if it is configured, backs up the metadata server (along with other resources) each Sunday at 1:00 a.m. by default.

  T I P To avoid conflicts with the weekly deployment backup, make sure not to schedule stand-alone metadata server backups for Sunday morning.

  - Unscheduled metadata server backups are launched automatically in certain situations, such as after the SAS Deployment Wizard configures the metadata server.

- Administrators can use SAS Management Console to change the metadata server backup schedule and configuration options, including the backup directory location and the backup retention policy. Backups can also be run on an ad hoc (unscheduled) basis from SAS Management Console, from the operating system command line, from SAS, or through third-party scheduling software.

- When repositories are restored in the event of a metadata server failure, transactions recorded in the journal file can be used to perform roll-forward recovery to a specified point in time after the last backup was taken.

- SAS Management Console displays detailed backup and recovery history. The history includes all backups that are run using the server's backup facility, regardless of how the backup was invoked (from the server's own scheduling facility, from SAS Management Console, from the command line, from SAS, or from third-party scheduling software).

- The backup process is a "hot" backup that executes in a separate thread while the metadata server is running. Therefore, the server does not need to be paused during the backup, and all transactions, including updates, continue to be processed. If journaling has been disabled or if the Reorganize Repositories backup option has been selected, the server is paused for Read-Only use, so that queries (but not updates) can continue to be processed.

- The metadata server runs its own backup and recovery processes. These processes do not rely on any other servers, such as workspace servers, DATA step batch servers, or scheduling servers. Because the metadata server has permission to access all of the needed files, there is no need to set up a host account with special operating system rights and permissions.

See Also
Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387

About the Automatically Configured Daily Backups of the Metadata Server

The SAS Deployment Wizard configures a schedule of automatic daily backups of the SAS Metadata Server. The automatic backups are configured as follows:

- Backups are performed at 1:00 a.m. server local time every day except Sunday.

- The backup that is performed on Monday uses the Reorganize Repositories option.

- Backups and their associated journal files are stored in the following location: *SAS-configuration-directory*/Lev1/SASMeta/MetadataServer/Backups. (If you select the Custom option when installing SAS 9.4, the SAS Deployment Wizard enables you to specify a different location.) Within the backup location, each set of backup files is stored in a directory that is named based on the date and time that the backup was started (for example, 2014-06-06T01_00_01-04_00). Do not

change this directory name, because the recovery facility requires the name to be in this format.

> **TIP** As a best practice, you should modify your backup configuration to specify a storage device other than the device that is used to store the metadata repositories and the server configuration files. Specifying a separate device will ensure that backup files and their associated journal files are available in the event of a disk failure.

> **TIP** Be sure to include this backup location in your regular system backups.

- Backups are retained for a period of seven days. Each time a backup is successfully completed, all previously created backup files that are more than seven days old are deleted from the backup directory. A record of the deleted backups is retained in backup history. (When a backup is attempted but not competed successfully, no backup files are deleted.)

The backup facility automatically backs up the following files:

- all of your metadata repositories, regardless of their access state (Online, Read-Only, Administration, or Offline).

- the repository manager.

- all of the files in *SAS-configuration-directory*/**Lev1/SASMeta/ MetadataServer**, including metadata server configuration files, INI files, SAS programs, scripts, and backup history files, as well as any additional files that you might have stored in this directory. These files are backed up to a **ConfigurationFiles** subdirectory within the backup location.

Note: If you have configured the Deployment Backup and Recovery Tool, it backs up the metadata server and other resources each Sunday at 1:00 a.m. by default. See Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

See Also

- "Modifying the Metadata Server Backup Schedule" on page 173
- "Modifying the Metadata Server Backup Configuration (Backup Location and Retention Policy)" on page 175

About Unscheduled Backups of the Metadata Server

In addition to running scheduled backups, the metadata server automatically backs itself up in certain unscheduled situations. Unscheduled backups use the same server-based backup facility and the same backup configuration options that are used for scheduled backups.

- A backup is run automatically after the SAS Deployment Wizard configures the SAS Metadata Server.

- A backup is run automatically after you complete a successful recovery of the metadata server. (Backups that are completed in this situation are not deleted when they are older than the retention period.)

- If you change the JOURNALTYPE option in the omaconfig.xml file to NONE or SINGLE (which is not recommended), and you later change the option back to ROLL_FORWARD, a metadata server backup is run automatically when you restart the server.

As a best practice, you should run an ad hoc backup before and after making other types of major changes to your system. For example, you should run a backup before and after the SAS Deployment Wizard configures the middle and client tiers of your deployment, before and after you change the host name of an existing server, before you delete a repository, and after you add a new repository.

See Also

-

-

How Journal Options Affect Metadata Server Backup and Recovery

The metadata server journal is a high-speed disk file in which transactions are recorded until they can be applied to repositories. The JOURNALTYPE option of the metadata server configuration file (omaconfig.xml) specifies whether metadata journaling is enabled and, if so, which type of journal file is created:

- The SAS Deployment Wizard sets the JOURNALTYPE option to ROLL_FORWARD. This option creates a linear journal file that permanently stores all transactions that have occurred since the most recent backup. The journal file is written to the same location as the associated backup files. Each time a new backup is executed, journaling stops and a new journal file is started in the new backup location.

  CAUTION:
  As a best practice, it is strongly recommended that you keep the default setting of ROLL_FORWARD for the JOURNALTYPE option.
  ROLL_FORWARD must be specified if you want to be able to use roll-forward recovery in the event of a metadata server failure.

- If you change the JOURNALTYPE option to SINGLE (which is not recommended), the roll-forward option is unavailable in the event of a metadata server failure. This setting creates a single circular journal file of a fixed size at the location that is specified in JOURNALPATH. Journal entries are appended to the file until the file grows to the size that is specified in JOURNALSIZE. Subsequent entries overwrite previously written entries starting at the beginning of the file.

- If you change the JOURNALTYPE option to NONE (which is not recommended), then no journal file is created and the roll-forward option is unavailable in the event of a metadata server failure. In addition, the metadata server cannot be backed up while the server is in an ONLINE state. The server will pause itself to a READONLY state each time a backup is executed, and any update transactions that are issued during the backup will fail.

See Also

-

-

About the Reorganize Repositories Backup Option

When metadata is deleted from a SAS metadata repository, the record is removed from both memory and disk. However, the disk space that had been allocated for the record remains in the data set. When you schedule or run a metadata server backup, you can select the Reorganize Repositories option to reclaim unused disk space left from the previously deleted metadata objects.

The default backup schedule specifies a weekly reorganization at 1:00 a.m. each Monday. It is not necessary to reorganize the repositories more frequently than once a week, except in extraordinary situations such as the deletion of a large amount of metadata.

If the Reorganize Repositories option is selected, the backup process does the following:

- pauses the server, placing it in a READONLY state

- copies the metadata server files to the backup destination

- re-creates the repository data sets in place, eliminating unused space in the process

- resumes the server to an ONLINE state

Note: The Reorganize Repositories option is ignored if it is used on a metadata server that has been started with clustering. However, you can use this option when backing up a single node that has been started without clustering. For detailed instructions, see "Start a Single Node without Clustering" on page 221.

CAUTION:
The Reorganize Repositories option should be used only during times of little or no user activity. The metadata server is paused during the reorganization process, and any update transactions that are issued during this process will fail.

SAS Metadata Server Backup Tasks

Credential Requirements for Administering Metadata Server Backups

To administer metadata server backups in SAS Management Console, you must be an unrestricted user, a user who is in the SAS Administrators group, or a user who is a member of the **Metadata Server: Operation** role. The tasks that require these credentials include modifying the backup schedule or the backup configuration, running ad hoc backups, and executing a metadata server recovery.

Modifying the Metadata Server Backup Schedule

If the default backup schedule for the metadata server is not appropriate for your site, you can change it using SAS Management Console. On the Plug-ins tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**. Then right-click **Server Backup** and select **Backup Schedule**.

The following dialog box appears:
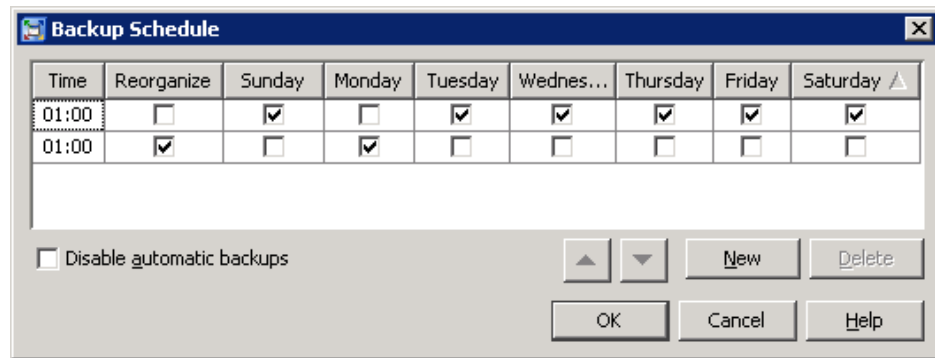
The dialog box displays a row for each time of day that backups are scheduled to run, with check marks indicating the days of the week. The **Time** column displays the time in a 24-hour time format; for example, `1:00` means 1:00 a.m., and `23:30` means 11:30 p.m. The backup runs when the local time of the metadata server host machine is equal to the specified time. Backups that include the Reorganize Repositories option are listed on separate lines with a check mark in the **Reorganize** column.

You can modify the schedule as follows:

- To change the time for an existing row in the schedule, select the **Time** field and enter the new time using a 24-hour format in server local time.

- To specify different days of the week in an existing row, select or clear the appropriate check boxes in the columns **Sunday** through **Saturday**.

- To specify whether the repositories are to be reorganized during the backups, select or clear the check box in the **Reorganize** column. In most cases, it is not necessary to reorganize the repositories more frequently than once a week. (This option is ignored if it is used on a metadata server that has been started with clustering. However, you can use this option when backing up a single node that has been started without clustering.)

- To add another row to the schedule, click **New**. Enter the time using a 24-hour time format in server local time. Then select the days of the week that the backup is to be run, and select the **Reorganize** option if needed. You must select at least one day on each line.

- To delete an entire row of the schedule, click any cell in the row and then click **Delete**.

- To prevent the scheduled backups from running, select the **Disable automatic backups** check box. The scheduled backups will not run until you clear this check box.

When you are finished, click **OK** to save the changes.

TIP If you schedule multiple backups per day, be sure to leave enough time for each backup job to complete before the next scheduled backup starts.

TIP If you have configured the Deployment Backup and Recovery Tool, remember that it backs up the metadata server and other resources each Sunday at 1:00 a.m. by default. To avoid conflicts, make sure not to schedule stand-alone metadata server backups for Sunday morning. See Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

Modifying the Metadata Server Backup Configuration (Backup Location and Retention Policy)

SAS Management Console enables you to modify the backup location as well as the backup retention policy. On the **Plug-ins** tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**. Then right-click **Server Backup** and select **Backup Configuration**.

The following dialog box appears:
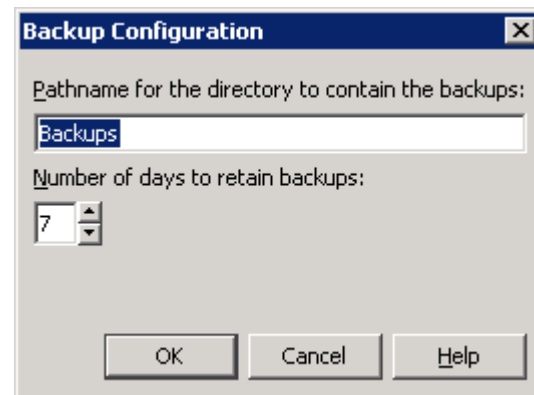
You can modify the configuration as follows:

- To change the backup location, specify either a complete path or a path relative to the MetadataServer directory. If the lowest-level directory in the path does not exist, it will be created. Directories that are higher in the path must already exist.

  T I P   As a best practice, you should specify a storage device other than the device that is used to store the metadata repositories and the server configuration files. Specifying a separate device will ensure that the backup files and their associated journal files are available in the event of a disk failure.

  Follow these guidelines:

  - You can specify any network location that is accessible from the metadata server. If you want to write the backup files to a path that is remote to the metadata server, then specify a UNC path (for example, `\\D2345\SASBackup`).

  - If your metadata server is installed on a Windows host, do not specify a mapped drive.

  - Make sure that the directory is specified as a shared directory and that the process owner for the metadata server has been granted full access. For clustered servers on Windows systems, make sure that the login account that you specified for the metadata server service has full access. To protect the integrity of the backups, you might want to deny access for other users.

  - Be sure to include the directory in your regular system backups.

- To change the number of days that backup files are retained in the backup location, select the appropriate number in the **Number of days to retain backups** selection box. Each time a successful backup is completed, previous backups that are older than the specified number of days are automatically deleted. The backup history automatically displays the offline status icon (⚠) for the deleted backups.

If you do not want backup files to be deleted automatically based on a retention policy, then select **0** in the selection box. If you make this selection, you will need to manually delete files from the backup location on a regular basis to ensure disk space availability.

Note: The offline status icon (⚠) is not displayed automatically for backups that you delete manually. To update the status icon for a manually deleted backup, you must open the backup's Properties dialog box.

When you are finished, click **OK** to save the changes.

Using SAS Management Console to Run an Ad Hoc Metadata Server Backup

After certain types of events, the metadata server automatically backs itself up. See "About Unscheduled Backups of the Metadata Server" on page 171. As a best practice, you should run an ad hoc backup before and after making other types of major changes to your system. For example, you should run a backup before and after the SAS Deployment Wizard configures the middle and client tiers of your deployment, before and after you change the host name of an existing server, before you delete a repository, and after you add a new repository.

Ad hoc backups use the same server-based backup facility and the same backup options that are used for scheduled backups. To run an ad hoc metadata server backup from SAS Management Console:

1. On the **Plug-ins** tab, navigate to **Environment Management ⇨ Metadata Manager ⇨ Metadata Utilities ⇨ Server Backup**.

2. Right-click **Server Backup** and select **Run Backup Now**.

3. In the dialog box that appears, you can enter a comment that will appear in the backup history; for example, you might want to state the reason for the ad hoc backup. (The comment is not displayed to clients of the metadata server.)

   You can also select the check box to **Reorganize repositories**.

   TIP The Reorganize Repositories option should be used only during times of little or no user activity. The metadata server is paused during the reorganization process, and any update transactions that are issued during this process will fail. This option is ignored if it is used on a metadata server that has been started with clustering.

The backup appears in backup history immediately after the backup process is complete.

Using the MetadataServer Command or backupServer.sas to Run Ad Hoc Metadata Server Backups

You can also run an ad hoc backup by using one of the following methods. Backups that are run using these methods use the same server-based backup facility and the same backup options that are used for scheduled backups.

- From the directory **SAS-configuration-directory/Lev1/SASMeta/ MetadataServer**, issue one of the following commands:

  - On Windows: **MetadataServer backup**

  - On UNIX or z/OS: **MetadataServer.sh backup**

  These commands execute the program backupServer.sas.

- Directly run the program backupServer.sas. The program is located in ***SAS-configuration-directory*/Lev1/SASMeta/MetadataServer**.

The backup appears in backup history in SAS Management Console's after you select **View ⇨ Refresh**. The following comment appears with the backup:

```
Started by user-ID with the MetadataServer command
```

Note: You cannot reorganize repositories when running a backup with the MetadataServer command or backupServer.sas.

> ***CAUTION:***
> **The backupServer.sas. program uses the metaparms.sas file, which contains the user ID and the encoded password for the SAS Administrator user. Be sure to protect this file from unauthorized access.**

Using Third-Party Scheduling Software to Run Metadata Server Backups

Instead of using the backup scheduling facility that is provided with the metadata server, you can choose to use a third-party scheduling system, as follows:

1. Use SAS Management Console to disable the metadata server's scheduled backups, as follows:

   a. On the **Plug-ins** tab, navigate to **Environment Management ⇨ Metadata Manager ⇨ Metadata Utilities ⇨ Server Backup**. Then right-click **Server Backup** and select **Backup Schedule**.

   b. In the Backup Schedule dialog box, select the **Disable automatic backups** check box, and click **OK**.

2. Set up your scheduling system to run the **MetadataServer backup** command or to directly execute backupServer.sas on a daily basis.

Troubleshooting Metadata Server Backup Jobs

If a backup does not finish successfully, an email with the subject line `Metadata Server Backup Failed` is sent to the designated recipients for metadata server alerts. The e-mail specifies the server's host name, the path to the server's configuration directory, and the backup name. For details about the failure, refer to the metadata server log and search for the backup name. The log is typically located in the Logs subdirectory of the server's configuration directory.

When a backup fails, an "x" icon (❌) also appears in the Status column of the backup and recovery history panel in SAS Management Console. To determine why the backup failed or why the backup files have become invalid, right-click the backup and select **Properties**. In the Properties dialog box, the **Error Summary** field provides high-level information to help resolve the problem. If more details are available, then the **Full Report** button is activated. Click this button to obtain a text report that you can browse, copy, and paste.

The ManifestFiles subdirectory of each backup contains files called MetadataServerBackupHistory.xml and MetadataServerBackupConfiguration.xml. In addition, the backup directory contains the MetadtaServerBackupManifest.xml file. These files might be useful to SAS Technical Support if you require troubleshooting assistance.

See Also
"Managing Alert Email Options for the SAS Metadata Server" on page 204

Recovering the SAS Metadata Server

About Metadata Server Recovery

You can use SAS Management Console to recover the SAS Metadata Server. The following options are available:

- You can recover just the metadata repositories and the repository manager, or you can choose to recover the configuration files as well.

- You can recover from a backup that is listed in the backup history panel, or you can recover from backup files that are stored in an alternate network-accessible location.

- You can use the roll-forward recovery to apply metadata updates that are stored in the journal file.

The recovery facility provides safeguards to ensure the integrity of the backup files from which you are recovering. The recovery operation checks to make sure that the backup directory contains all of the correct files and that the files have the correct name and file sizes. In addition, each backup file contains a universal unique identifier that is used to make sure that you are recovering files for the correct metadata server. If any problems exist, the recovery is not started and a warning message is displayed.

During recovery operations, the metadata server is automatically paused to a RECOVERY state. This state is similar to an OFFLINE state but more restrictive. External systems such as Windows Services Manager report the server as paused. After the recovery, the metadata server automatically takes a new backup. If the recovery was successful, the server is automatically returned to the state that it was in before the recovery process.

Note: In SAS 9.4M1, the metadata server script includes a –recover option. This option starts a server that is not currently running, and then restores the server's metadata repository from the most recent backup. The option provides an easy way to recover a server or node that is unresponsive. The option does not provide roll-forward recovery, recovery of configuration files, or recovery from a backup other than the most recent backup. See "Additional Script Options for the SAS Metadata Server" on page 74 and "Recovering a Clustered Metadata Server" on page 223.

Note: You cannot use SAS Management Console to execute a recovery on a metadata server that has been started with clustering. To recover a clustered metadata server, see "Recovering a Clustered Metadata Server" on page 223.

See Also
Chapter 13, "What to Do If the SAS Metadata Server Is Unresponsive," on page 185

About Roll-Forward Recovery

When recovering the metadata server using SAS Management Console, you can choose to use roll-forward recovery to update the recovered metadata repository files with transactions from the journal file. Roll-forward recovery uses the journal file that is located in the same directory as the backup files. The updates that are applied include additions, modifications, and deletions of metadata, as well as additions and deletions of

metadata repositories. You can choose to apply all transactions that are in the journal file, or you can apply all transactions through a specified date and time.

Roll-forward recovery can be used only if the JOURNALTYPE option of the metadata server configuration file (omaconfig.xml) was set to ROLL_FORWARD at the time that the metadata server updates were made. (The ROLL_FORWARD option is enabled automatically by the SAS Deployment Wizard.)

CAUTION:
> **If you recover the metadata server without selecting the roll-forward option, then any update or delete transactions that occurred after the backup was taken will be lost. In addition, any repositories that were after the backup was taken will be deleted.**

Executing a Metadata Server Recovery

To use SAS Management Console to recover the metadata server from a backup, either with or without roll-forward recovery, follow these steps:

1. Log on to SAS Management Console as an unrestricted user, a user who is in the SAS Administrators group, or a user who is a member of the **Metadata Server: Operation** role. On the **Plug-ins** tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**.

2. As a best practice, it is recommended that you pause the server to the ADMINISTRATION state.

   If you start the recovery while the server in the ADMINISTRATION state, the server changes the state to RECOVERY, performs the recovery, and then returns the server to the ADMINISTRATION state. You can then verify system readiness before clients reconnect. When readiness has been verified, you can manually resume the server.

   If you start the recovery while the server is ONLINE, the recovery process pauses the server to the RECOVERY state and then returns it to the ONLINE state after a successful recovery.

3. Use one of the following methods to start the recovery process:

   • If you want to recover using backup files that are in the configured backup location, follow these steps:

     1. On the backup history list in the right panel, find the backup from which you want to recover. To determine its status, either double-click the backup entry or right-click and select **Properties**. If the files are usable for a recovery, the **Verification** field of the Properties dialog box displays this message: `All files in this backup are currently correct.`

        You can also review the detailed information about the backup on the **General** and **Repositories** tabs.

     2. When you are sure that you have identified the correct backup, right-click the backup entry in the backup history and select **Recover from This Backup**.

   • If you want to recover from a different location (for example, the location that contains your system backups), right-click the **Server Backup** node (located on the **Plug-ins** tab under **Metadata Utilities**), and select **Recover from Alternate Location**. In the Recover from Alternate Location dialog box, enter the complete pathname for the directory that contains the backup. You can specify any network location that is accessible from the metadata server. If the backups are remote to the metadata server, then specify a UNC path, as in the following example:

`\\D2345\SASMetadataServer\2011-01-14T01_00_01-05_00`

Note: If your metadata server is installed on a Windows host, do not specify a mapped drive.

4. After you select **Recover from This Backup** (or after you enter a pathname in the Recover from Alternate Location dialog box and click **OK**), the server accesses the backup directory and attempts to validate the backup files. It checks to make sure that all of the backup files are present and that they have the correct filenames, file sizes, and universal unique identifier.

Note: If the validation is not successful, one of the following messages is displayed:

- `A Recover cannot be run because this Backup is offline and not available.` If this message appears, you will need to recover from a different set of backup files.

- `This Backup does not match what was backed up. Are you sure you want to recover from this Backup?` If this message appears, select **Details** to determine which files in the backup directory are incorrect.

  If the configuration file backups are incorrect but the repository backups are correct, you can click **OK** to resume the recovery. However, make sure that you have not selected the option to **Also recover server configuration files**.

  If the repository backups are incorrect, click **Cancel** and select a different backup from which to recover.

If the validation is successful (or if you resumed the recovery after receiving an error message), the Recover from This Backup dialog box appears:

5. In the first text field of the Recover from This Backup dialog box, enter a comment that will appear in the backup history panel with this recovery. For example, you might want to state the reason that the server is being recovered.

6. In the second text field, enter a comment that will be displayed to users while the server is paused during the recovery.

7. If appropriate, select the check boxes for the following options:

Also recover server configuration files
This option does the following:

- deletes all files from the server's configuration directory (for example, *SAS-configuration-directory*/**Lev1/SASMeta/MetadataServer**)

- from the **ConfigurationFiles** subdirectory in the backup location, copies all files to the server's configuration directory

CAUTION:

If you select "Also recover server configuration files," you will lose any additions or modifications to files in the MetadataServer directory that have been made since the backup was taken. As a best practice, you should select this option only if your configuration files have become corrupted.

Reorganize recovered repositories
This option reorganizes the metadata repositories when they are backed up following the recovery. The reorganization reclaims any unused disk space that is left from previously deleted metadata objects. See "About the Reorganize Repositories Backup Option" on page 173.

Roll forward transactions
In addition to restoring the repository files from the backup, this option recovers transactions that were recorded in the metadata server journal file after the backup was created. If you select this option, the following additional options become available:

All transactions in journal file
This option recovers all transactions that are recorded in the journal file. The time period that is covered by the journal file is displayed on the right side of the dialog box in server local time. (If you want to display the time period in Greenwich Mean Time (GMT), select **Show time values in GMT**.)

Transactions through date/time
This option recovers all transactions through a date and time that you specify. This option is useful if you want to recover the server to the state that it was in just before a specific problematic transaction. You can use the metadata server log to determine the exact time that the transaction occurred. See "Use the Log Tab in the Server Manager" on page 119.

Enter the roll-forward ending date and time in the format *yyyy-mm-dd***T***hh:mm:ss*. You can either edit the default values that are displayed in the field, or you can cut and paste the date and time from the appropriate event in the metadata server log. Most log events use server local time.

The dialog box enables you to specify the time in either server local time or GMT, depending on which of the following radio buttons is selected:

- **Show time values in server local time** displays the journal file's default starting and ending time in server local time. You can override the ending date and time. Be sure to enter the time in server local time if this radio button is selected.

- **Show time values in GMT** displays the journal file's default starting and ending times in Greenwich Mean Time. You can override the ending date and time. Be sure to enter the time in GMT if this radio button is selected.

CAUTION:

> **The preceding radio buttons do not convert your entry to server local time or to GMT. If you select or deselect one of these radio buttons after entering a date and time, your entry remains unchanged.**

8. Before starting the recovery operation, the server pauses itself to a RECOVERY state. This state is similar to an OFFLINE state but more restrictive.

   Note: For example, the RECOVERY state prevents clients from obtaining a list of registered repositories. When the server is in the RECOVERY state, external systems such as Windows Services Manager report the server as paused.

9. When the recovery is completed, a backup of the newly recovered server is taken automatically, regardless of whether the recovery completed successfully.

   When the post-recovery backup is completed, all previous backups are retained (that is, no backups are deleted based on the retention policy).

10. If the recovery completed without errors, the server is returned to the state that it was in prior to the recovery process.

    If a recovery fails, the server is placed in the ADMINISTRATION state. The administrator can then access SAS Management Console and try to resolve the errors or recover the server from a different backup.

Troubleshooting Metadata Server Recovery Jobs

If a recovery does not finish successfully, an email with the subject line Metadata Server Recovery Failed is sent to the designated recipients for metadata server alerts. The e-mail specifies the server's host name, the path to the server's configuration directory, and the recovery name. For details about the failure, refer to the metadata server log and search for the recovery name. The log is typically located in the **Logs** subdirectory of the server's configuration directory.

When a recovery fails, an "x" icon () also appears in the Status column of the backup and recovery history panel in SAS Management Console. To determine why the recovery failed, right-click the recovery and select **Properties**. On the Properties dialog box, the **Error Summary** field provides high-level information to help resolve the problem. If more details are available, then the **Full Report** button is activated. Click this button to obtain a text report that you can browse, copy, and paste.

The **ConfigurationFiles** subdirectory of each backup contains files called MetadataServerBackupHistory.xml, MetadataServerBackupConfiguration.xml and MetadataServerRecoveryManifest.xml. These files might be useful to SAS Technical Support if you require troubleshooting assistance.

Note: In a Windows environment, the recovery operation cannot update a READONLY file. If you have manually set any of the metadata server files to READONLY, recovery errors will be reported. In this situation, the recovery might still be valid if no changes have been made to the READONLY file since the backup was taken.

See Also
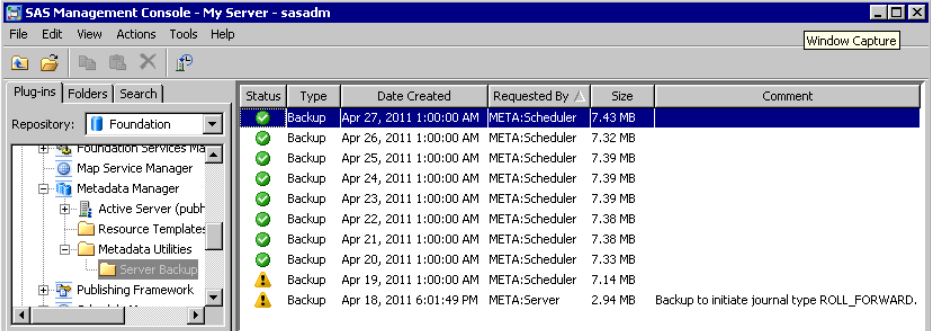"Managing Alert Email Options for the SAS Metadata Server" on page 204

Viewing Backup and Recovery History

SAS Management Console provides a list of all backups and recoveries that have been run using the server's backup and recovery facility. All backups performed by this facility are listed regardless of how the process was invoked (from the server's own scheduling facility, from SAS Management Console, from the command line, from SAS, or from third-party scheduling software).

To view the backup and recovery history list:

1. Log on to SAS Management Console. On the **Plug-ins** tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**.

2. Select the **Server Backup** node. The backup and recovery history appears in the right pane, as shown in the following example:

CAUTION:

> **The icons that are initially displayed do not necessarily report the most recent status of a backup. To obtain the most recent status, you must right-click the backup and select Properties.**

Note: If you migrated to SAS 9.4 from a previous SAS version, the backup history does not display any backup history prior to the migration.

3. To obtain more information about a backup or a recovery, and to update the status icon for a backup, right-click the history entry and select **Properties**. The Properties dialog box appears, displaying a text description of the status and other information about the backup. Explanatory error messages are displayed if the backup was unsuccessful or if it is found to be invalid.

Icons in the history list indicate the most recently reported status for each backup or recovery. The status icons have the following meanings:

- The checkmark icon ( ) means that the backup or recovery was successful. For backups, this icon also means that the backup was determined to be valid the last time the files were checked. A backup is considered valid if all of the files are present in the backup location, all of the files have the correct GUID, and all filenames and file sizes are correct.

- The exclamation point icon means that the backup was successful, but the backup files are currently offline (that is, they are not present in the backup location). The files might have been deleted automatically based on the retention policy, or

they might have been moved or deleted manually. This icon appears automatically when backups are deleted based on the retention policy.

- The "x" icon ✖ indicates one of the following situations:
  - the backup or recovery was not successful
  - the backup was successful, but when the files were last checked, they were found to be invalid (that is, one or more files is missing, or one or more files has an incorrect GUID, incorrect name, or incorrect size)

The properties dialog box provides details about the backup or recovery, as shown in the following example:
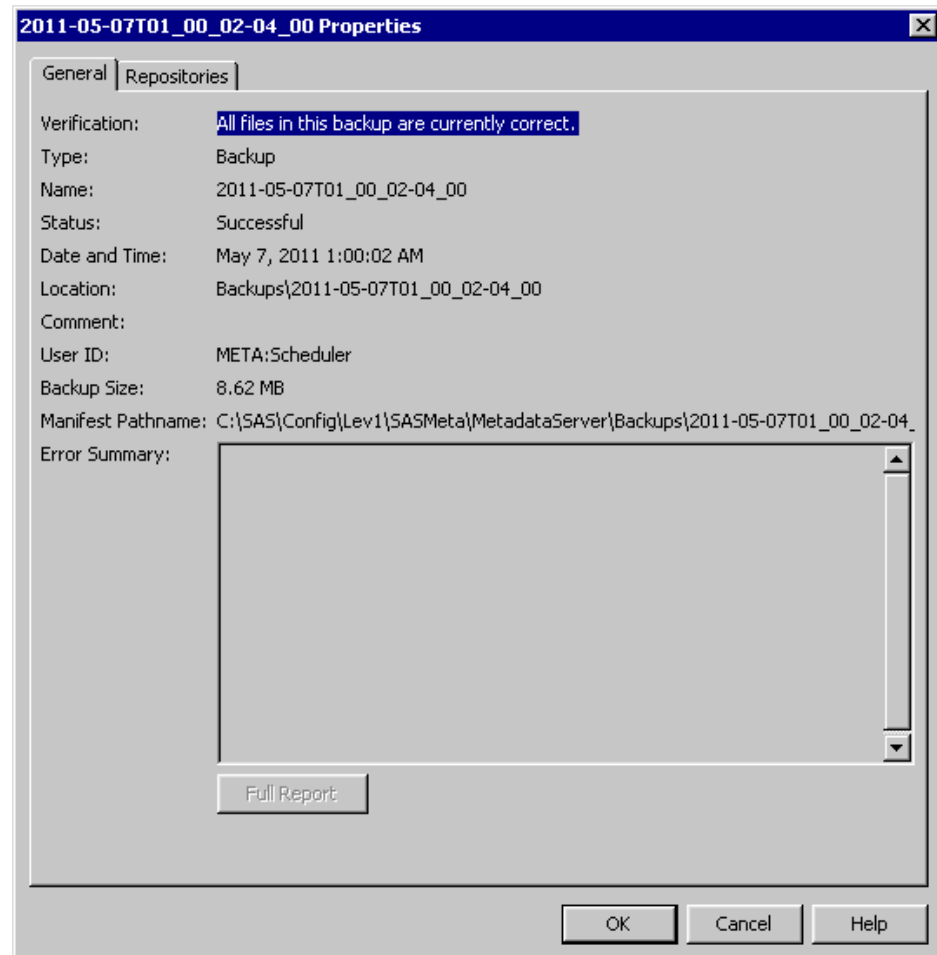
The **Repositories** tab lists the repositories that were included in the backup or recovery. (The **Repositories** tab does not appear if the backup is offline. It also does not appear for recoveries other than the most recent recovery.)

The Error Summary field displays any errors that occurred during the backup or recovery or during the verification of a backup. If more details are available, then the **Full Report** button is activated. Click this button to obtain a text report that you can browse, copy, and paste.

Chapter 13

What to Do If the SAS Metadata Server Is Unresponsive

Recovering an Unresponsive SAS Metadata Server: Initial Steps

If the metadata server is not responding to client requests, follow these steps:

1. Use one of the following methods to stop the metadata server. In this situation, these are the preferred methods for stopping the server.

    • Use the Metadata Manager in SAS Management Console. Select the **Plug-ins** tab, expand the **Metadata Manager** node, right-click the **Active Server** node, and select **Stop**.

Or you can use the metadata server script, as described in "Use Individual Scripts to Operate Servers" on page 70.

- (Windows only) If you cannot stop the server using the Metadata Manager or the script, then stop the Windows service. If you cannot stop the service, then use the Windows Task Manager to stop the server process.

- (UNIX only) If you cannot stop the server using the Metadata Manager or the script, use one of the following commands to stop the server process:

  ```
  kill -2 server-process-id
  ```

  ```
  kill -15 server-process-id
  ```

  If the process fails to stop, use the following command:

  ```
  kill -9 server-process-id
  ```

  For more information, see "Exiting or Interrupting Your SAS Session in UNIX Environments" in the *SAS Companion for UNIX Environments*.

  Verify that the process has successfully stopped.

2. Review the entries in the metadata server log to determine the cause of the problem. For example, the log might indicate an issue in your operating environment.

   By default, the log is located in **SAS-configuration-directory/Lev1/ SASMeta/MetadataServer/Logs**. Your site might have selected the option to write all server logs to a single directory, such as **SAS-configuration-directory/Lev1/Logs**.

   If the log identifies an issue in your environment, resolve the issue, if possible.

3. Start the server as described in Chapter 6, "Operating Your Servers," on page 61.

4. If you still cannot start the server, review the log again and resolve any issues, if possible.

5. If you are unable to resolve the issue and start the server, you will need to use manual steps to start it. You will then be able to use the metadata server recovery facility to restore the server from a backup. See the following topics:

   - "Recover an Unresponsive Metadata Server (Non-clustered)" on page 186
   - "Recovering an Unresponsive Clustered Metadata Server" on page 192

   CAUTION:
   If the server cannot be started, do not attempt to re-initialize the server or perform a recovery without referring to the procedures in one of these topics.

Recover an Unresponsive Metadata Server (Non-clustered)

Overview

Complete the following steps to recover a non-clustered metadata server that is not responding to client requests:

1. Make a copy of the server's current data and configuration.

2. Try to start the server using the repair flag in the omaconfig.xml file. If the server starts successfully, then remove the repair flag and stop the server. Restart the server with the original omaconfig.xml file. Further steps are not necessary.

3. Try to start the server using the -recover option. If the server starts successfully, skip to step 5.

4. If the server does not start successfully after you complete step 3 , then manually recover the configuration files.

5. Use SAS Management Console to perform a normal recovery. You must perform this step after completing step 4.

   If the server started successfully after step 3 , then complete this step only if you want to roll forward transactions or you want to restore from a backup other than the most recent backup.

6. Run the Analyze and Repair Metadata tool.

Details are provided in the following subtopics.

Step 1 (Non-clustered Server): Copy Current Configuration Information and Data

Before starting to recover an unresponsive server, it is important to make a copy of the server's current data and configuration. This copy is needed if you request assistance from SAS Technical Support. Follow these steps:

1. Make sure that the metadata server is stopped.

2. Open the metadata server start-up directory (`SAS-configuration-directory/Levn/SASMeta/MetadataServer`). Then use operating system commands to make copies of all the files and subdirectories in the `/SASMeta/MetadataServer` directory.

Step 2 (Non-clustered Server): Try to Start the Server Using the Repair Flag in the omaconfig.xml File

1. Add the `<DEBUG REPAIR_METADATA_FILES ="y"/>` flag to the omaconfig.xml file and try to start the metadata server with the `DEBUG REPAIR_METADATA_FILES` option as described in "Repair and Analyze Metadata Files Options" on page 457.

2. Remove the `<DEBUG REPAIR_METADATA_FILES="y"/>` flag from the omaconfig.xml file.

   If the metadata server still fails to start, continue to the next step.

   If the metadata server starts, then stop the metadata server and start it again normally.

Step 3 (Non-clustered Server): Try to Start the Server Using the -recover Option

1. Go to the metadata server start-up directory (`/SASMeta/MetadataServer`), and issue the following command:

- Windows: **MetadataServer.bat -recover**

- UNIX: **MetadataServer.sh -recover**

  Note: The -recover option became available in SAS 9.4M1.

  If the server starts successfully, the -recover option restores the metadata repository from the most recent backup.

2. If the server does not start, complete "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188.

   If the server starts successfully, skip "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188. Instead, take one of the following actions:

   - If you want to perform roll-forward recovery to a specified point in time after the backup was taken, or if you want to restore a prior version of the metadata, then complete "Step 5 (Non-clustered Server): Use SAS Management Console to Perform a Normal Recovery" on page 190 and "Step 6 (Non-clustered Server): Run the Analyze and Repair Metadata Tool" on page 192.

   - If you do not need to roll forward or restore a previous version of the metadata, then go directly to "Step 6 (Non-clustered Server): Run the Analyze and Repair Metadata Tool" on page 192.

Step 4 (Non-clustered Server): Manually Recover the Configuration Files

Follow this procedure only if you were unable to start the server using the instructions in "Step 3 (Non-clustered Server): Try to Start the Server Using the -recover Option" on page 187.

1. Go to the metadata server start-up directory (*SAS-configuration-directory*/**Levn/SASMeta/MetadataServer**), and do the following:

   a. Make sure that the **/SASMeta/MetadataServer** subdirectory contains the following subdirectories: **Logs**, **rposmgr**, **Backups**, and **MetadataRepositories**. If these directories do not exist, then create them.

   b. Use operating system commands to delete the following:

      - any subdirectories and files that are present in **/SASMeta/MetadataServer/rposmgr**. (Do not delete the directory.)

      - any subdirectories and files that are present in **/SASMeta/MetadataServer/MetadataRepositories**. (Do not delete the directory.)

2. Locate the directory that contains the most recent metadata server backup. The backup facility names backup directories based on the date and time that the backup was started. For example, a directory called **2015-06-30T01_00_01-04_00** contains files from a backup that was started just after 1:00 a.m. on June 30, 2015. The backup directory should contain the following:

   - a subdirectory for each metadata repository

   - a subdirectory called **ConfigurationFiles**

   - a subdirectory called **ManifestFiles**

   - the following files: MetadataServerBackupManifest.xml and MetadataServerJournal.dat

3. Copy the contents of the **ConfigurationFiles** subdirectory of the backup
    directory (for example, **2015-06-30T01_00_01-04_00/**
    ConfigurationFiles) to the metadata server start-up directory (**/SASMeta/**
    MetadataServer), as illustrated here:.

Copy From	What to Copy	Copy To

4. Copy the contents of the **ManifestFiles** subdirectory of the backup directory (for
    example, **2015-06-30T01_00_01-04_00/ManifestFiles**) to the **Backups**
    subdirectory of the metadata server start-up directory (**/SASMeta/**
    MetadataServer), as illustrated here. The directory should contain the files
    MetadataServerBackupConfiguration.xml and MetadataServerBackupHistory.xml.

Copy From	What to Copy	Copy To

5. Copy the file MetadataServerBackupManifest.xml from the backup directory (for
    example, **2015-06-30T01_00_01-04_00**) to the metadata server start-up
    directory (**/SASMeta/MetadataServer**), as illustrated here:

Copy From	What to Copy	Copy To

Copy only the file MetadataServerBackupManifest.xml.

6. Use a text editor to open the file adminUsers.txt (located in **/SASMeta/ MetadataServer**). This file is used to define the metadata server's unrestricted users.

   Add an external user (that is, a user that is recognized as a user of the server's operating environment) to the file using the following format: ***user-ID**. Be sure to place the asterisk in the first column. In Windows environments, the ID must be qualified (for example: ***domain\user-ID**).

   This temporary change is used in "Step 5 (Non-clustered Server): Use SAS Management Console to Perform a Normal Recovery" on page 190. The change is rolled back once it is no longer needed. For more details about the adminUsers.txt file, see "Overview of Configuration Files" on page 17.

7. Start only the metadata server.

   On UNIX, use the MetadataServer.sh -start command. (Do not use the sas.servers script to start all of your SAS processes.) On Windows, start the metadata server Windows service. See Chapter 6, "Operating Your Servers," on page 61.

8. If the server starts successfully, you must now complete "Step 5 (Non-clustered Server): Use SAS Management Console to Perform a Normal Recovery" on page 190.

Step 5 (Non-clustered Server): Use SAS Management Console to Perform a Normal Recovery

Complete this step in either of the following situations:

- You have completed "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188. Now you must use SAS Management Console to recover the repository data sets and the repository manager.

- The server started successfully in "Step 3 (Non-clustered Server): Try to Start the Server Using the -recover Option" on page 187. If necessary, you can now use SAS Management Console to perform roll-forward recovery to a specified point in time after the backup was taken, or to restore a prior version of the metadata.

Follow these steps:

1. Log on to SAS Management Console. If you completed "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188, then log on with the external user ID that you created in step 6 of that procedure.

   If warning messages appear, respond as follows:

Message	Response
`Connection Warning: "Connecting to a server that is paused for administration."`	**OK**
`No Repositories Found: "No Repositories are present on the requested server. Would you like to add or register a repository now?"`	**No**
`Error: "The connection to the SAS Metadata Server was broken."`	**OK**

   When SAS Management Console opens, make sure that the correct domain and user ID appear in the lower right corner of the window.

2. On the **Plug-ins** tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**.

3. In the Backup History panel, locate the backup from which you want to recover. Right-click the backup entry, and select **Recover from this backup**.

   Note: If the backup does not appear in the backup history list, then right-click the **Server Backup** node on the **Plug-ins** tab, and select **Recover from alternate location**. (You must use this method if you followed the instructions in "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188.)

   Note: The recovery operation checks the backup files to ensure that they are valid. If a warning message appears, you might need to select a different backup to restore.

4. In the Recover from This Backup window, you can specify the roll-forward option to restore the maximum amount of SAS metadata. If the metadata server log indicates that the server was corrupted by a SAS metadata transaction, then be sure to roll forward to a point in time prior to the problem transaction. For details about this and other recovery options, see "Recovering the SAS Metadata Server" on page 178.

5. If you added an external unrestricted user to adminUsers.txt while completing "Step 4 (Non-clustered Server): Manually Recover the Configuration Files" on page 188, it is recommended that you either delete that user from the file or restore adminUsers.txt from a backup. You must then restart the metadata server to apply the change.

Step 6 (Non-clustered Server): Run the Analyze and Repair Metadata Tool

When the recovery process is complete, it is recommended that you use the Analyze and Repair Metadata tool before giving users access to the server. This tool is located in the Metadata Manager in SAS Management Console. For details, see "Analyzing and Repairing Metadata" on page 236.

Note: When running this tool after a recovery, you must close SAS Management Console and log on to it again as an unrestricted user (for example, sasadm@saspw).

Recovering an Unresponsive Clustered Metadata Server

Use the following procedure if a clustered metadata server is offline because all of the nodes are non-responsive or a quorum of the nodes are non-responsive.

Step 1 (Clustered Server): Copy Current Configuration Information and Data

Before starting to recover an unresponsive server, it is important to make a copy of the server's current data and configuration. This copy is needed if you request assistance from SAS Technical Support. Follow these steps:

1. Make sure that the metadata server is stopped.

2. Open the metadata server start-up directory (**SAS-configuration-directory/Levn/SASMeta/MetadataServer**). Then use operating system commands to make copies of all the files and subdirectories in the **/SASMeta/MetadataServer** directory.

Step 2 (Clustered Server): Determine Which Node Was the Most Recent Controller Metadata Server

On each node in the cluster, check the metadata server logs for the following string: "Redirect client in cluster SASMeta - Logical Metadata Server".

This string indicates that the node is acting as the controller metadata server and has the most recent metadata updates in the repository.

Step 3 (Clustered Server): Use the omaconfig.xml Repair Flag to Start the Most Recent Controller Metadata Server

On the most recent metadata server identified in step 2, add the **<DEBUG REPAIR_METADATA_FILES ="y"/>** flag to the omaconfig.xml file as described in "Repair and Analyze Metadata Files Options" on page 457.

Example of usage in omaconfig.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<OMAconfig>
```

```
<DEBUG REPAIR_METADATA_FILES="y"/>
```

Then, try to start the metadata server with **`MetadataServer.sh -startNoCluster`**.

If the metadata server still fails to start, then remove the **`<DEBUG REPAIR_METADATA_FILES ="y"/>`** flag from the omaconfig.xml file and continue to the next step.

If the metadata server starts, then stop the metadata server, remove the **`<DEBUG REPAIR_METADATA_FILES="y"/>`** flag from the omaconfig.xml file, and start it again with **`MetadataServer.sh -startNoCluster`** using the original omaconfig.xml file. The node is started in Administration state. Continue to

Step 4 (Clustered Server): Use the -recover Option to Start the Initially Configured Node

1. Make sure that all of the nodes in the metadata server cluster have been stopped. Use the following command:

    *   Windows: **`MetadataServer.bat -stopCluster`**

    *   UNIX: **`MetadataServer.sh -stopCluster`**

2. On the initially configured node (that is, the node that was initially configured by the SAS Deployment Wizard), go to the metadata server start-up directory (***`SAS-configuration-directory`*`/Levn/SASMeta/MetadataServer`**). Issue the following command:

    *   Windows: **`MetadataServer.bat -recover`**

    *   UNIX: **`MetadataServer.sh -recover`**

    Note: The -recover option became available in SAS 9.4M1. This option starts a node that is not currently running, and then restores the node's metadata repository from the most recent backup. The node is then paused to offline status as it waits for other nodes to join the cluster.

3. If you want to perform roll-forward recovery to a specified point in time after the most recent backup was taken, or if you want to restore a prior version of the metadata, then complete

    If you do not need to roll forward or restore a previous version of the metadata, then go to

Note: The recovery process for a clustered server does not include recovering configuration files from the backup. Recovery of these files is not recommended for clustered servers, since the backed up files could contain node-specific paths or options.

Step 5 (Clustered Server): Use SAS Management Console to Perform a Normal Recovery

Complete this step if both of the following are true:

*   You have successfully completed

- You want to perform roll-forward recovery to a specified point in time after the backup was taken, or you want to restore a prior version of the metadata.

Follow these steps:

1. Use the following command to stop the node that you recovered in "Step 4 (Clustered Server): Use the -recover Option to Start the Initially Configured Node" on page 193:

   - Windows: **MetadataServer.bat -stop**

   - UNIX: **MetadataServer.sh -stop**

2. Use the following command to restart the node in non-clustered mode:

   - Windows: **MetadataServer.bat -startNoCluster**

   - UNIX: **MetadataServer.sh -startNoCluster**

   The node is started in Administration state.

3. Log on to SAS Management Console. On the **Plug-ins** tab, navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨ **Server Backup**.

4. In the Backup History panel, locate the backup from which you want to recover. Right-click the backup entry, and select **Recover from this backup**.

   Note: If the backup does not appear in the backup history list, then right-click the **Server Backup** node on the **Plug-ins** tab, and select **Recover from alternate location**.

   Note: The recovery operation checks the backup files to ensure that they are valid. If a warning message appears, you might need to select a different backup to restore.

5. In the Recover from This Backup window, you can specify the roll-forward option to restore the maximum amount of SAS metadata. If the metadata server log indicates that the server was corrupted by a SAS metadata transaction, then be sure to roll forward to a point in time prior to the problem transaction. For details about this and other recovery options, see "Recovering the SAS Metadata Server" on page 178.

Step 6 (Clustered Server): Run the Analyze and Repair Metadata Tool

If you just completed the recovery process in step 5 on page 193, it is recommended that you use the Analyze and Repair Metadata tool before giving users access to the server. This tool is located in the Metadata Manager in SAS Management Console. For details, see "Analyzing and Repairing Metadata" on page 236.

Note: When running this tool after a recovery, you must close SAS Management Console and log on to it again.

If you did not perform a recovery and were directed to this step from step 4, complete these steps:

1. Use the following command to stop the node that you recovered in "Step 4 (Clustered Server): Use the -recover Option to Start the Initially Configured Node" on page 193:

   - Windows: **MetadataServer.bat -stop**

   - UNIX: **MetadataServer.sh -stop**

2. Use the following command to restart the node in non-clustered mode:

   • Windows: **MetadataServer.bat -startNoCluster**

   • UNIX: **MetadataServer.sh -startNoCluster**

   The node is started in Administration state.

3. Run the Analyze and Repair Metadata tool. This tool is located in the Metadata Manager in SAS Management Console. For details, see "Analyzing and Repairing Metadata" on page 236.

Step 7 (Clustered Server): Back Up the Node and Restart It in Cluster Mode

It is important to complete this backup step. The backup that is taken is used in Step 5 to recover the remaining nodes in the cluster. Follow these steps:

1. Log on to SAS Management Console. On the **Plug-ins** tab, navigate to **Environment Management ⇨ Metadata Manager ⇨ Metadata Utilities ⇨ Server Backup ⇨ Run Backup Now**.

2. Use the following command to stop the node that you recovered in "Step 5 (Clustered Server): Use SAS Management Console to Perform a Normal Recovery" on page 193:

   • Windows: **MetadataServer.bat -stop**

   • UNIX: **MetadataServer.sh -stop**

3. Use the following command to restart the node in clustered mode:

   • Windows: **MetadataServer.bat -start**

   • UNIX: **MetadataServer.sh -start**

Step 8 (Clustered Server): Recover the Remaining Nodes in the Cluster

On each of the remaining nodes in the cluster, go to the metadata server start-up directory (**/SASMeta/MetadataServer**), and issue the following command:

• Windows: **MetadataServer.bat -recover**

• UNIX: **MetadataServer.sh -recover**

After the node is started and its repository has been restored, the node rejoins the cluster and its repository is updated to match the other nodes in the cluster.

Repeat this procedure on each remaining node. When a quorum of nodes has been recovered, the server is online and available to process client requests.

The following message should appear in the metadata server log: The cluster has achieved quorum and is now ONLINE. You can also use SAS Management Console to view the cluster's status. Select the **Plug-ins** tab, expand the **Metadata Manager** node, right-click the **Active Server** node, and select **Properties**. On the Properties window, select the **Cluster** tab.

Part 5

Administering the SAS Metadata Server

Chapter 14

Metadata Server Configuration Tasks

Overview of SAS Metadata Server Configuration

The following configuration settings can be used to manage the SAS Metadata Server:

- the MAXACTIVETHREADS, THREADSMIN, and THREADSMAX options, which control the way that the metadata server uses threads. See "Configuring the Number of Threads Used by the Metadata Server" on page 200.

- the metadata server journaling options. See "Configuring Metadata Server Journaling" on page 202.

- the alert email options. See "Managing Alert Email Options for the SAS Metadata Server" on page 204.

Configuring the Number of Threads Used by the Metadata Server

About the MAXACTIVETHREADS, THREADSMIN, and THREADSMAX Options

When the metadata server is initialized, the server creates a pool of available threads to run client requests. The metadata server's configuration settings control the minimum and maximum number of threads in the pool.

When a client request is received, one of the available threads is used to process the request. If no threads are available, the server creates a new thread. If the thread pool has reached the maximum size, the request waits until a thread is returned to the pool.

The client request is then executed in the thread as resources become available. The server's configuration settings control the maximum number of threads that can be executed concurrently. When processing is complete, the thread is returned to the pool. If the number of available threads exceeds the minimum thread pool size, unused threads are destroyed after a period of time to conserve system resources.

The management and use of threads is controlled by the following configuration settings:

minimum number of threads
determines the minimum number of threads that are maintained in the server's thread pool, regardless of the number of client requests that are received or accepted. If the number of available threads exceeds this minimum, then the excess unused threads are destroyed after a period of time to conserve system resources.

By default, the server uses this formula to compute the minimum number of threads:
`MAX(5,((number-of-processors * 2) + 1))`

This means that the value is set to either 5 or to ((*number-of-processors* * 2) + 1), whichever is greater.

For example:

- If the host machine has eight processors, then the minimum number of threads is set to 17.

- If the host machine has two processors, then the minimum number of threads is set to 5.

- If the host machine has one processor, then the minimum number of threads is set to 5.

If the metadata server's invocation options include a value for THREADSMIN, then that value is used instead of the default.

maximum number of threads
determines the maximum number of threads that are maintained in the server's thread pool, regardless of the number of client requests that are received or accepted. If no threads are available when a client request is received, then a new thread is created only if the number of the threads in the thread pool is less than the maximum. If the maximum has already been reached, the client request waits for a thread to be returned to the pool.

As a best practice for the metadata server, THREADSMAX and THREADSMIN should be set to the same value to avoid thread resource thrashing.

By default, if no value is specified for maximum number of threads, it is set to the computed value of THREADSMIN.

maximum number of active threads
determines the number of threads that are allowed to run concurrently on the metadata server. This number controls the number of queries that can be active on the server at any one time.

By default, the server sets the maximum number of active threads as follows:

- If the host machine has one processor, then the maximum active number of threads is set to 2.

- If the host machine has two or more processors, then the maximum active number of threads is set to the number of processors.

If the metadata server's configuration file includes a value for MAXACTIVETHREADS, then that value is used instead of the default.

The default settings that are described in the preceding list provide a good starting point for most sites. To fine-tune the configuration to meet your specific workload and performance requirements, you can set different values in your configuration files. For instructions and guidance, see the next topic.

Specify Values for the MAXACTIVETHREADS, THREADSMIN, and THREADSMAX Options

To fine-tune the performance of the metadata server, you can specify values for the following:

- the MAXACTIVETHREADS option in the metadata server configuration file, omaconfig.xml

- the THREADSMIN and THREADSMAX parameters of the OBJECTSERVERPARMS metadata server invocation option

The values that you specify override the default configuration settings that are described in "About the MAXACTIVETHREADS, THREADSMIN, and THREADSMAX Options" on page 200.

The following are some examples of situations in which you might want to adjust these settings. You might encounter other situations in which changes are beneficial.

- If you want to dedicate only a portion of the host machine's processors to metadata server processing, then specify values for MAXACTIVETHREADS, THREADSMIN, and THREADSMAX that are lower than the default settings. The following settings are recommended as a starting point:

  - If you want to dedicate more than one processor to metadata server processing, then set MAXACTIVETHREADS to the number of processors that you want to use. For example, if your host machine has eight processors, but you want to dedicate only five of those processors to metadata server processing, then set MAXACTIVETHREADS to 5.

    Note: As a best practice, you should not set MAXACTIVETHREADS to a value less than 2.

  - After you determine the correct setting for MAXACTIVETHREADS, specify a value for THREADSMIN and THREADSMAX that is equal to (MAXACTIVETHREADS * 2) + 1. For example, if you set

MAXACTIVETHREADS to 5, then set THREADSMIN and THREADSMAX to 11.

- If clients of the metadata server are experiencing processing delays, and if you have observed that some of the processors that are dedicated to the server are underutilized, then you might want to specify values for MAXACTIVETHREADS, THREADSMIN, and THREADSMAX that are higher than the default settings. Follow these guidelines:

  - As a general rule, THREADSMIN and THREADSMAX should be set to a value that equals (MAXACTIVETHREADS * 2) + 1.

  - Increase the settings in small increments. For example, you could first set MAXACTIVETHREADS to *number-of-processors* + 2 and set THREADSMIN and THREADSMAX to a value that equals (MAXACTIVETHREADS * 2) + 1.

    After each change, check to see whether processor utilization increases and delay times decrease. If so, gradually increase the settings to determine which level results in the best performance. (When the settings are too high, the performance benefits will diminish.)

To specify a value for MAXACTIVETHREADS, update the metadata configuration file omaconfig.xml. See "Reference Information for omaconfig.xml" on page 456. To specify values for THREADSMAX and THREADSMIN, update the metadata server's sasv9_usermods.cfg file. See "Modifying the Metadata Server Invocation Options" on page 469. Both of these files are located in the following path:

`SAS-configuration-directory`/Lev1/SASMeta/MetadataServer

After modifying the configuration files, you must restart the metadata server to enable the changes. The server reads the configuration settings only during initialization.

Configuring Metadata Server Journaling

About Metadata Server Journaling

Journaling is enabled by default for the SAS Metadata Server. For best performance, it is recommended that journaling be enabled at all times. In addition, journaling must be properly configured in order for roll-forward recovery to be available in the event that you need to restore the metadata server.

The SAS Metadata Server is an "in-memory" server. As clients submit queries and updates, the requested records are read from repository data sets into the server's memory. When journaling is not enabled, clients that request metadata updates must wait for updates to be written to both the in-memory records and the repository data sets on the file system before client access is resumed.

When journaling is enabled, access is returned to clients as soon as the metadata updates are written to the in-memory database and the journal file. The more time-consuming updates to the repository data sets are performed later in the background. If problems occur that prevent the repository data sets from being updated from the journal, an alert email message is generated. If the metadata server fails before the update process has had a chance to apply all updates from the journal file, the metadata server automatically recovers them from the journal file when it is restarted.

The metadata server is initially set up to write journal entries to **`SAS-configuration-directory`/Lev1/SASMeta/MetadataServer/Journal**. The journal file in this location contains all updates that have been made to metadata since

the most recent backup was taken. The next time the server is backed up, the following actions are taken:

- The active journal file is closed, copied to the directory that contains the associated backup, and renamed to MetadataServerJournal.dat, as in this example: ***SAS-configuration-directory*/Lev1/SASMeta/MetadataServer/Backups/ 2015-07-15T01_00_00-04_00/MetadataServerJournal.dat**.

- A new journal file is created in the **Journal** directory with a filename that includes the name of the new backup. For example: ***SAS-configuration-directory*/ Lev1/SASMeta/MetadataServer/Journal/ 2015-07-15T01_00_00-04_00_Journal.dat**.

See Also
["Managing Alert Email Options for the SAS Metadata Server" on page 204](#)

About the Journal Options

Journaling is controlled by options that are set in the omaconfig.xml file when you install or migrate your system. In most cases, there is no need to change the initial settings.

The OMA JOURNALTYPE option specifies the type of journal file that is created. By default, the SAS Deployment Wizard sets the value of this option to ROLL_FORWARD, which creates a linear journal file that permanently stores all transactions that have occurred since the most recent backup. The journal file is written to the same location as the associated backup files. Each time a new backup is executed, journaling stops and a new journal file is started in the new backup location.

CAUTION:
ROLL_FORWARD must be specified if you want to be able to use roll-forward recovery in the event of a metadata server failure.

Other allowable values for OMA JOURNALTYPE are SINGLE and NONE. These values are not recommended. If you specify one of these values, then roll-forward recovery is not available.

Additional options in omaconfig.xml that apply to journaling include the following:

- JOURNALSIZE and JOURNALPATH, which are applicable only if you specify SINGLE as the OMAJOURNALTYPE.

- JOURNALYIELDCOUNT, which specifies the number of records that the update process can apply before yielding to allow other internal processing tasks. Do not change this setting without consulting with SAS Technical Support.

See Also
- ["Reference Information for omaconfig.xml" on page 456](#)
- ["How Journal Options Affect Metadata Server Backup and Recovery" on page 172](#)

Managing Alert Email Options for the SAS Metadata Server

About Metadata Server Alerts

The metadata server sends alert emails in the following situations:

- An error occurs during a metadata server backup or recovery.

- A problem occurs that prevents the repository data sets from being updated from the journal.

- One of the worker nodes in a clustered server fails to initialize or synchronize and shuts itself down.

Alert emails that are generated by the metadata server are sent to the addresses that are specified in the OMA ALERTEMAIL option in the omaconfig.xml file. The generated email has `Metadata Server Alert` in the subject line. The body of the message specifies the error that occurred, the name of the metadata server host machine, the metadata server port, and the location of the metadata server log.

Effective with SAS 9.4M2, reminder emails are sent every six hours if the alert condition is not corrected. If the alert condition persists after three days, the server shuts down automatically to avoid additional issues.

How Alert Email Options for the Metadata Server Are Configured

To generate alert emails, the metadata server refers to the following configuration options:

- OMA ALERTEMAIL in omaconfig.xml specifies one or more email addresses to which alerts are to be sent.

- EMAILSYS, EMAILHOST, and EMAILPORT options in the metadata server's sasv9.cfg file provide information about the email protocol and email server.

- The EMAILID option in the metadata server's sasv9.cfg file provides information that appears in the From address of alert emails. If the email server requires authentication, this option also provides the user ID that is used for authentication.

- The EMAILAUTHPROTOCOL and EMAILPASSWORD options in the metadata server's sasv9.cfg file provide additional information to authenticate to the email server, if authentication is required.

- EMAILSSL in omaconfig.xml specifies whether the email server is configured with TLS (SSL). For more information, see "<OMA> Options".

- EMAILACKWAIT in the metadata server's sasv9.cfg file specifies how long that SAS waits to receive an acknowledgment from the SMTP server. For more information, see EMAILACKWAIT System Option.

- SSLCALISTLOC in the metadata server's sasv9.cfg file specifies the location of the public certificate(s) for the trusted certificate authorities (CA). For more information, see SSLCALISTLOC System Option.

The initial settings for these options are based on information that is specified in the SAS Deployment Wizard. Values for EMAILAUTHPROTOCOL, EMAILID, and

EMAILPASSWORD can be specified in the wizard only if the custom installation option is selected.

See Also

- To update alert email addresses, see "Reference Information for omaconfig.xml" on page 456.

- To modify email server or authentication information, see "Modifying the Metadata Server Invocation Options" on page 469.

- To update the email user ID or password, see "About Updating Managed Passwords" in *SAS Intelligence Platform: Security Administration Guide*.

Testing Email Alerts for the SAS Metadata Server

After your SAS system has been initially deployed, you should run a test to make sure that alert emails are working properly. You should repeat this test after you make any changes to your alert email configuration.

To test email alerts for the metadata server:

1. Log on to SAS Management Console as a user who has the appropriate credentials. (See "Who Can Do What in the Metadata Manager Plug-in" on page 40.)

2. In the **Plug-ins** tab, expand the **Metadata Manager** node, right-click the **Active Server** node, and select **Properties**.

3. In the Active Server Properties dialog box, select **Send Test Message**.

4. In the Send Alert E-mail Test Message dialog box, enter text to be included in the email. Then select **OK**.

The metadata server sends an email message to the addresses that are specified in the OMA ALERTEMAIL option in the omaconfig.xml file. The generated email has `Metadata Server Alert Email Test` in the subject line. The body includes the text that you entered in the dialog box, followed by the name of the metadata server host machine, the metadata server port, and the location of the metadata server log.

If the test is successful, the following message is displayed: `An alert e-mail message was sent to` *address*. If the test is not successful, an explanatory error message is displayed.

You can also use the OMA ALERTEMAILTEST option in the omaconfig.xml file to test alert emails. However, this method requires stopping the server.

Chapter 15

Managing Metadata Server Memory

Setting the Server's MEMSIZE Parameter

The SAS Metadata Server holds all queried and updated metadata in memory. Therefore, to avoid excessive paging, the amount of RAM on the server should exceed the amount of metadata.

To ensure that sufficient memory is available on Windows and UNIX systems, set a MEMSIZE value in the start server command that allows the operating system to set the memory limits on the server. This can be accomplished by setting a value of 0 (zero), MIN, or MAX. For details, see Chapter 32, "Options for the Metadata Server Invocation Command," on page 469.

Input and Output Balancing

Because all metadata is held in memory, and because all updates to the metadata are single-threaded, you do not have to configure input/output contentions. Queries cause I/O activity only when the metadata is initially loaded into memory based on type. For example, when a user requests metadata for a specific column, metadata for all columns is loaded into memory. No additional I/O for column metadata queries occur until the server is paused and resumed or stopped and restarted.

Configuring Concurrent Queries and Clients

The number of processors that are available to the metadata server determines the number of concurrent queries that can run on the server.

If you expect a small volume of concurrent clients, you can probably use a single-processor machine. If you are expecting 1,000 concurrent clients, you will want an eight-processor machine.

By default, the number of queries (threads) that can be processed concurrently is set as follows:

• If the host machine has one processor, then the maximum number of active threads is set to 2.

• If the host machine has two or more processors, then the maximum number of active threads is set to the number of processors.

You can allow a different maximum number of active threads by specifying a value for the server's MAXACTIVETHREADS option. You can also configure the number of threads that are available in the server's thread pool, or you can change the configuration so that only a portion of the host machine's processors are dedicated to metadata server processing. For details, see "Configuring the Number of Threads Used by the Metadata Server" on page 200.

Using optimizeIMDB to Optimize and Report on Metadata Server Memory Usage

About the Metadata Server optimizeIMDB Command Option

To optimize the usage of memory on the metadata server, you should occasionally use the optimizeIMDB command option of the metadata server script. This option analyzes the handling of character variables, checking for items such as unnecessary indexes, long character variables that can be stored as variable-length strings, and duplicate strings. A report is produced showing the expected amount of memory that will be required for each repository after the optimizations are applied. The optimizations are then applied the next time you pause and resume (or restart) the metadata server.

Note: You cannot use the optimizeIMDB option on a metadata server that has been started with clustering. However, you can use it on a single node that has been started without clustering.

When to Optimize Memory Usage

It is not necessary to perform memory optimization on a regular basis. Variable usage is unlikely to change much unless you add or delete a significant percentage of the total metadata in a repository. Optimization is most useful after metadata is first loaded into SAS Metadata Repositories and after significant updates are applied.

How Memory Optimization Works

The following processing occurs when you use the optimizeIMDB command option:

- The subprogram accesses the repository data sets in the backup destination. In each data set, it analyzes the handling of character variables. The analysis checks for such items as unnecessary indexes, long character variables that can be stored as variable-length strings, and duplicate strings.

- If the subprogram identifies optimizations for a repository, it stores the results of the analysis in a data set called DS_VCHAR. This data set is stored in the directory that contains the repository files (for example, **MetadataRepositories/ Foundation**) and in the directory that contains the most recent backup of the repository files.

- The next time the metadata server is paused and resumed or restarted, it merges the information in each repository's DS_VCHAR data set with the repository's METAIMDB data set. The updated information in the METAIMDB data set is then used to optimize memory usage. The optimizations are not applied to repository data sets on disk.

  After the changes are applied, the DS_VCHAR data sets are deleted from the repository directories (but not from the backup directories).

  CAUTION:
  The metadata server recovery facility deletes the DS_VCHAR data sets from the backup directories before it restores the repositories to their registered locations. Therefore, the optimizations are lost if you run a metadata server recovery before pausing and resuming (or restarting) the server.

Running the optimizeIMDB Command Option

You should be aware of the following considerations before using the optimizeIMDB command option:

- Because optimizeIMDB analyzes backup copies of your metadata repositories, a backup copy of the repositories must have been created by using the metadata server backup facility.

- If your backup directory is on the same machine as the metadata server, then you might want to run optimizeIMDB during a period of low activity to minimize the impact on client applications.

- The backup facility creates the OMA_Varan.sas7bcat catalog that is used by optimizeIMDB command option. The information in OMA_Varan.sas7bcat reflects the last backup that was made. Therefore, optimizeIMDB analyzes all repositories that were registered at the time of the backup.

- Because optimizeIMDB does not interact directly with the metadata server, no special user IDs or server privileges are required. However, the user executing the command needs full operating system access to the SAS Metadata Server backup destination path and all subdirectories.

To run the optimizeIMDB command option:

1. If you have configured a clustered metadata server, stop all of the nodes in the cluster. Then start a single node without clustering. See "Start a Single Node without Clustering" on page 221.

2. Open a command prompt and navigate to the metadata server start-up directory, which is in the following path: **SAS-configuration-directory/Lev1/ SASMeta/MetadataServer**.

3. Type one of the following commands:

   • On Windows: **MetadataServer -optimizeIMDB**

   • On UNIX or z/OS: **MetadataServer.sh -optimizeIMDB**

4. When the command is finished, you can review the following reports that are written to the backup directory:

   • Repository_memory_usage_on_*date and time*.html, which shows the amount of memory that is expected to be required for each repository after the optimizations are applied.

   • reports called *repository_id*_metaimdb_changes.txt, which list the specific memory optimizations that were identified for each repository

5. Be sure to pause and resume (or restart) the metadata server to apply the results of the analysis.

   If you ran the tool on a clustered metadata server node, you must restart (or stop and start) the node to place it in cluster mode as the controller node. Then start the other nodes in the cluster. The controller node applies the results of the analysis to the other nodes.

Recovering Memory on the SAS Metadata Server

The SAS Metadata Server supports two approaches for quickly recovering memory on a running server:

• Issue a Pause action followed by a Resume action. The Pause action instructs the SAS Metadata Server to stop accepting new client requests and closes the repository containers. The Resume action resumes client activity. Repository containers are re-opened as clients request metadata from them.

• Issue a Reset action. The Reset action pauses and resumes the metadata server in one step.

Pause, Resume, and Reset actions can be issued in SAS Management Console. See "Use SAS Management Console to Operate the SAS Metadata Server" on page 76. Pause and Resume actions can also be issued by using PROC METAOPERATE. See *SAS Language Interfaces to Metadata*.

Chapter 16
Using Metadata Server Clustering

About Metadata Server Clustering

SAS 9.4 provides the option of implementing a metadata server cluster. This feature provides redundancy and high availability of the metadata server, which is a core component of the SAS infrastructure. Clustering ensures that the server continues to operate if a server host machine fails.

A metadata server cluster is a group of three or more host machines (nodes) that have been configured as identical metadata servers. Each node runs its own server process and has its own server configuration information, journal file, and copy of the repository data sets. In addition, each node maintains a complete in-memory copy of the metadata repository. The nodes work together as if they were a single metadata server.

Client applications and users interact with the cluster in the same way that they would interact with a metadata server that is not clustered. A load-balancing process automatically distributes work among the nodes. If a node ceases to operate, the server continues to be available using the remaining nodes.

Effective with the SAS 9.4M2, metadata server clustering is supported on z/OS. In previous SAS 9.4 releases, it is supported only on UNIX and Windows.

Metadata server cluster machines should meet the requirements in the System Requirements for SAS®9.4 Foundation for their respective operating systems. In addition, the machines should also meet the following requirements:

1. All of the nodes in the cluster must have the same operating system and version.

2. All of the nodes in the cluster must be at the same SAS hot fix level.

3. Systems must be responsive:

    •   A clustered metadata server is a stateful application and each of the systems and the network should be robust and responsive to ensure cluster stability.

    •   Metadata server nodes should be stopped while intensive snapshot maintenance is performed.

    •   Virtual machines that are quiesced, or paused to a non-responsive state, will cause problems in the metadata cluster.

    •   Static resourcing for virtual machines that host metadata server nodes is recommended.

How Metadata Server Clustering Works

When a clustered metadata server is started, the nodes establish communication with one another. One of the nodes becomes the controller node, coordinating activity within the cluster. The other nodes are considered worker nodes.

When a client connects to the metadata server, a load-balancing process redirects the connection to one of the available nodes. This node evaluates the client's request to determine whether an update to metadata is required. The client's request is then processed as follows:

•   If the request does not require an update to metadata, the node immediately executes the request using metadata that is stored on the node.

•   If the request does require an update to metadata, the node returns the request to the controller node. The controller node processes the update and then immediately propagates the metadata changes to all of the worker nodes in the cluster.

If the controller node fails, another node is immediately promoted to serve as the controller node, and the cluster resumes operation. If a worker node fails, it drops out of the cluster, an email alert is generated, and load balancing uses only the remaining worker nodes. These failover processes occur automatically, with no need for user or administrator intervention and no noticeable effect on user activity.

For a cluster to continue to operate, a quorum of the nodes must be running. If fewer than a quorum is running, the server is paused to offline status. Clients can continue to connect to the server, but they cannot access metadata.

In clusters with an odd number of nodes, a quorum exists if more than one half of the nodes are running. For example, in a three-node cluster, two running nodes constitute a

quorum. In a five-node cluster, three running nodes constitute a quorum. The following table provides example scenarios for a three-node cluster.

Table 16.1   *Quorum Determination in a Cluster with an Odd Number of Nodes*

Node 1 (Initially Configured Server)	Node 2	Node 3	Quorum?	Server (Cluster) Status
●	●	●	Yes	Online
●	●	■	Yes	Online
●	■	■	No	Offline
■	●	●	Yes	Online
■	●	■	No	Offline

In clusters with an even number of nodes, one half of the nodes constitutes a quorum if one of the running nodes is the initially configured server. For example, in a four-node cluster, two nodes constitute a quorum if one of the running nodes is the initially configured server. If the initial server is not running, then all of the other three nodes must be running to constitute a quorum. The following table provides example scenarios for a four-node cluster.

Table 16.2   *Quorum Determination in a Cluster with an Even Number of Nodes*

Node 1 (Initially Configured Server)	Node 2	Node 3	Node 4	Quorum?	Server (Cluster) Status
●	●	●	●	Yes	Online
●	●	●	■	Yes	Online
●	●	■	■	Yes	Online
●	■	■	■	No	Offline
■	●	●	●	Yes	Online
■	●	●	■	No	Offline
■	●	■	■	No	Offline

The status of each node and the overall status of the cluster (including the presence or absence of a quorum) is reported in SAS Management Console on the **Cluster** tab of the

Active Server Properties window. Expand the **Metadata Manager** node, right-click the **Active Server** node, select **Properties**, and select the **Cluster** tab.

Note: If your clustered SAS Metadata Server becomes unresponsive and reports that the quorum is maintained, the SAS Metadata Server might have lost communication with one of its nodes. See this Problem Note for more information and a possible workaround.

Configuring a Metadata Server Cluster

Prerequisites for Cluster Configuration

The following are prerequisites for setting up metadata server clustering:

- You must use a deployment plan that specifies a multiple-machine deployment.

- All of the host machines in the cluster must have the same operating system.

- Each host machine must meet the usual specifications for a metadata server. For example, each machine must have as much memory as would be required for a non-clustered server.

- Storage space for metadata server backups must exist in a location that all of the host machines in the cluster can access using the same network path.

- You must specify the external user account that will start the nodes, as follows:

  - On Windows, the nodes are started by the SAS Metadata Server: Service Login Account. A suggested user ID is sassvlgn. The SAS Deployment Wizard prompts you for this account on each node. Specify an external account that is recognized by all of the machines that will participate in the cluster. For example, the account could exist in a Windows domain in which all of the machines participate.

    Be sure to grant this user Full Control of the `SASMetata\MetadataServer` directory on each node.

    Note: If you are using Integrated Windows authentication (IWA) for this account, see "IWA to a Clustered Metadata Server" in *SAS Intelligence Platform: Security Administration Guide*.

  - On UNIX and z/OS, the nodes are started by the SAS Installer account. The SAS Deployment Wizard prompts you for this account on each node.

  - On all operating systems, make sure that the account has Read and Write access to the metadata server's shared backup location. In addition, make sure that you specify the same account on all of the metadata server nodes.

- Assigned host names for metadata server nodes must be accessible from all of the other nodes in the cluster. If you use a HOSTS file entry to override the host name on one of the nodes, be sure to make the same entry in the HOSTS file on each of the other nodes.

- If your server tier or middle tier includes one or more Windows hosts, then you must specify an external account that will be used to run the Deployment Backup and Recovery tool. This account is referred to as the backup user. The account must meet the requirements that are specified in "What Is a Backup User?" on page 397.

- If a SAS Metadata Server cluster is installed in a virtual machine environment, ensure that the virtual machines are configured adequately to ensure a robust computing environment.

  - Do not use the VMware 'quiescence flag'.

  - Do not use Vmotion except during maintenance periods.

  This is because when the VMware pauses the virtual machines to perform snapshots or if Vmotion causes a system pause while the virtual machine (VM) is being moved to a different host system, then this can affect the metadata server's execution and the cluster stability.

  For more information about the VMware 'quiescence flag', see Understanding VM snapshots in ESXi. For more information about Vmotion, see VMware Vmotion.

Overview of Configuring a Metadata Server Cluster

You can configure metadata server clustering during the initial installation and configuration process for SAS 9.4. Alternatively, you can convert an existing SAS 9.4 metadata server to a cluster at any time by adding two or more additional nodes.

Review the prerequisites for metadata server clustering. Then follow these steps to configure the cluster:

1. Install and configure the initial metadata server.

   If the server has already been configured, you might need to modify its configuration to prepare for clustering.

2. Install and configure the additional metadata server nodes.

3. Update the metadata cluster profile on all applicable servers on page 218.

Step 1: Configure the Initial Metadata Server

Use the SAS Deployment Wizard to Install and Configure the Initial Metadata Server

Install and configure the initial metadata server in the same way that you would install and configure a non-clustered server. When you run the SAS Deployment Wizard, respond to the cluster-related prompts as specified in the following table:

Page in SAS Deployment Wizard	Response
SAS Metadata Server: Override Backup Location	Select **Override the default SAS Metadata Server backup directory**.
SAS Metadata Server: Backup Location	Specify a backup location that all of the nodes in the cluster are able to access using the same network path. The location should be highly available and must be separate from the initial metadata server machine and the additional cluster nodes.
SAS Metadata Server: Override Service Login Account (Windows only)	Select the check box **Specify the login account for the SAS Metadata Server**.

Page in SAS Deployment Wizard	Response
SAS Metadata Server: Service Login Account (Windows only)	Specify the external account that will be used to start the server (for example, *my-domain*\sassvlgn). This must be the same account that you will specify to start the other nodes. See "Prerequisites for Cluster Configuration" on page 214.
	This user is automatically granted **Log on as a service** rights if the user does not already have them.
External Account: Installer	Specify external account credentials for the SAS Installer account. On UNIX machines, this account is used to start the server.
SAS Deployment Backup and Recovery Tool: User Account (Windows only) *Note:* This prompt should appear during installation of the middle tier. *Note:* If this prompt does not appear, then you must configure the user manually after the deployment process is complete. For instructions, see "Specifying a Backup User Manually" on page 398.	Specify the external account that will be used to run the Deployment Backup and Recovery tool (for example, *my-domain*\sasbackup). This account is referred to as the backup user. The account must meet the requirements that are specified in "What Is a Backup User?" on page 397.

Modify the Configuration of an Existing Metadata Server to Prepare for Clustering

If the initial metadata server has already been configured, you might need to take the following steps to meet the prerequisites for server clustering:

1. Ensure that the configured backup location is a location that all of the machines in the cluster can access using the same network path. The location should be highly available and must be separate from the initial metadata server machine and the additional cluster nodes. Follow these steps if you need to change the configured location:

   a. In SAS Management Console, select **Metadata Manager** ➪ **Metadata Utilities** ➪ **Server Backup** ➪ **Backup Configuration**.

   b. In the Backup Configuration window, specify the network pathname.

2. Ensure that the server is started with an external account that is recognized by all of the machines that will participate in the cluster. On Windows systems, follow these steps if you need to configure a different account:

   a. Stop the metadata server.

   b. In the Windows Services Manager, right-click the **SASMeta - Metadata Server** service and select **Properties**.

   c. On the **Log On** tab of the Properties window, specify the information for the appropriate external account. See "Prerequisites for Cluster Configuration" on page 214.

   d. Restart the metadata server.

3. If your server tier or middle tier includes one or more Windows hosts, then ensure that you have specified a backup user to run the Deployment Backup and Recovery tool. For details, including instructions for defining the user manually, see "Specifying a Backup User to Access a Central Vault Location or a Clustered Metadata Server (Windows Only)" on page 397.

Step 2: Install and Configure Additional Metadata Server Nodes

To configure each additional node for the metadata server cluster, do the following:

1. Start the SAS Deployment Wizard on the machine that will host the new node. Make sure that the machine meets the prerequisites that are described in "Prerequisites for Cluster Configuration" on page 214.

2. As you run the SAS Deployment Wizard, respond to the cluster-related prompts as specified in the following table:

Page in SAS Deployment Wizard	Response
Select Deployment Task	Select **Install SAS software**.
Select Deployment Type	Select **Perform a Planned Deployment**, **Install SAS software**, and **Configure SAS software**.
Specify Deployment Plan	Specify the deployment plan for your installation.
Select Deployment Step and Products to Install	Select **Step 2: Metadata Server Node**.
Select Configuration Prompting Level	Select **Typical**.
Specify Configuration Information	Specify the configuration directory path and configuration level for the new node. As a best practice, it is recommended that you specify the same directory path and configuration level (for example, Lev1) on all of the nodes.
Local Machine Name	Specify the fully qualified host name and short host name for the machine on which you are installing the node.
SAS Metadata Server	Specify the fully qualified host name and other connection information for the initial (currently running) metadata server.

Page in SAS Deployment Wizard	Response
SAS Metadata Server: Service Login Account (Windows only)	Specify the external account that will be used to start the server. This must be the same user account that is used to start the other nodes (including the initially installed metadata server).
	This user is automatically granted **Log on as a service** rights if the user does not already have them.
Local Machine Port	Specify the local machine port that the new metadata server node uses. As a best practice, it is recommended that you specify the same port (for example, 8561) on all of the nodes.
External Account: Installer	Specify external account credentials for the SAS Installer account. On UNIX machines, this account is used to start the server. As a best practice, this should be the same SAS Installer account that was specified for the initial metadata server.

3. When the installation and configuration process is complete, use the Server Manager in SAS Management Console to connect to the node and validate the configuration. For details, see "Monitoring Clustered Metadata Servers" on page 222.

4. Repeat the steps 1 through 2 on each machine that you want to add to the cluster.

5. Restart the middle-tier servers and the required applications in order to read the latest metadata cluster configuration information.

6. Restart the middle-tier servers a second time to update the web applications with this new information.

Step 3: Update Metadata Cluster Profile on All Applicable Servers

Use sas-update-metadata-profile to update the metadata profiles in your SAS Application Server's configuration with information from the cluster definition. The command should be run on each machine that contains an object spawner, a SAS/CONNECT spawner, or components of SAS Application Servers (such as workspace servers, pooled workspace servers, OLAP servers, and stored process servers). For details, see "Update Metadata Profiles (sas-update-metadata-profile)" on page 381.

Modifying an Existing Metadata Server Cluster

Add a Node to an Existing Cluster

To add a node to an existing metadata cluster, follow the instructions in "Step 2: Install and Configure Additional Metadata Server Nodes" on page 217.

Remove a Node from an Existing Cluster

If you need to remove a node from a cluster configuration, use the Remove Existing Configuration feature of the SAS Deployment Manager. For instructions, see "Removing a SAS Configuration" in *SAS Intelligence Platform: Installation and Configuration Guide.*.

The server continues to operate as a cluster as long as your configuration includes the initially configured metadata server and at least two additional nodes.

If your configuration includes only the initially configured server and one additional node, the server continues to operate. However, it will not operate as a cluster, and therefore it will not provide any redundancy or failover support.

Note: You cannot remove the initially configured metadata server unless you are removing all components that depend on the metadata server.

Convert a Clustered Configuration to a Nonclustered Configuration

To convert a clustered configuration to a nonclustered configuration, use SAS Deployment Manager to remove the configuration for each of the nodes (except the initially configured metadata server). See "Remove a Configuration" in *SAS Intelligence Platform: Installation and Configuration Guide*.

Managing a Clustered Metadata Server

Backing Up a Clustered Metadata Server

If you have implemented metadata server clustering, you use the metadata server backup facility in the same way that you would use it for a server that is not clustered. The facility backs up the node that is currently acting as the controller node. It is not necessary to back up the other nodes, because they are exact copies of the controller node.

In the backup configuration for each node, make sure that you have specified the same backup destination. This destination must be accessible to all of the nodes via the same network path. This ensures that the backup works regardless of which node is acting as the controller node.

The Reorganize Repositories (REORG) backup option is ignored when you back up a server that has been started as a clustered server. However, you can use this option when backing up a single node that has been started without clustering. For detailed instructions, see "Start a Single Node without Clustering" on page 221.

Updating Server Configuration Files and Scripts on a Clustered Metadata Server

If you change a configuration file or start-up script that is associated with the metadata server, be sure to make the identical changes on each node in the cluster. This rule applies to files in the **SASMeta/MetadataServer** directory such as omaconfig.xml, logconfig.xml, adminUsers.txt, sasv9_usermods.cfg, and MetadataServer_usermods.bat or MetadataServer_usermods.sh.

Synchronizing Metadata on a Clustered Metadata Server

SAS 9.4M3 includes a new metadata analysis and repair tool called Metadata Server Cluster Synchronization. This tool verifies that metadata is synchronized among all the nodes of a metadata server cluster. If metadata differences are found, the repair option sets a flag on the server, writes messages about the issue to the Analyze and Repair log file, and generates an alert email. The next time the problem node is restarted, the flag causes the node's metadata to be synchronized with metadata on the controller node.

You might want to run this tool daily or weekly to make sure your metadata is in sync. The tool is also useful after you recover from the failure of a node, or if you need to troubleshoot inconsistent metadata query results.

For details, see "Analyzing and Repairing Metadata" on page 236.

Using Integrated Windows Authentication to a Clustered Metadata Server

Clustered metadata servers support Integrated Windows Authentication (IWA) for connections from SAS desktop clients. However, you must manually register service principal names (SPNs). For details, see "IWA to a Clustered Metadata Server" in the *SAS Intelligence Platform: Security Administration Guide*.

Starting, Stopping, Pausing, and Resuming a Clustered Metadata Server

Start a Clustered Metadata Server

To start a clustered metadata server, use your preferred method (for example, the server script, SAS Environment Manager, or Windows Service Manager) to start each of the nodes in the cluster. The nodes can be started in any order.

Make sure that all of the nodes are running before you start components that depend on the metadata server. These components include objects spawners, OLAP servers, SAS/CONNECT spawners, and SAS/SHARE servers.

Stop a Clustered Metadata Server

To stop a clustered metadata server, use one of the following methods:

- In SAS Management Console, select **Metadata Manager ⇨ Active Server ⇨ Stop**.

- On any one of the nodes in the cluster, use the following command. All of the nodes in the cluster are stopped automatically.

  - On Windows:

    ```
    MetadataServer.bat -stopCluster
    ```

  - On UNIX:

    ```
    MetadataServer.sh -stopCluster
    ```

Alternatively, you can use your preferred method (for example, the server script, SAS Management Console, SAS Environment Manager, or Windows Service Manager) to stop each of the nodes individually.

Pause a Clustered Metadata Server

To pause a clustered metadata server, select **Metadata Manager** ⇨ **Active Server** ⇨ **Pause** in SAS Management Console. Then select the state to which the server is to be paused (**Administration**, **Offline**, or **Read Only**).

Alternatively, you can use your preferred method (for example, the server script, SAS Management Console, SAS Environment Manager, or Windows Service Manager) to pause any one of the nodes in the metadata server cluster. When you pause a node, the entire cluster is paused to OFFLINE status.

Resume a Clustered Metadata Server

To resume a clustered metadata server, select **Metadata Manager** ⇨ **Active Server** ⇨ **Resume** in SAS Management Console.

Alternatively, you can use your preferred method (for example, the server script, SAS Management Console, SAS Environment Manager, or Windows Service Manager) to resume any one of the nodes in the cluster. When you resume a node, the entire cluster is resumed.

Start a Single Node without Clustering

To start a single node without clustering, use the following command:

- On Windows:

  ```
  MetadataServer.bat -startNoCluster
  ```

- On UNIX:

  ```
  MetadataServer.sh -startNoCluster
  ```

The node starts as a single, non-clustered metadata server that is paused to the Administration state. This action is useful when you want to perform one of the following administrative tasks on a node:

- perform a metadata server recovery

- back up the metadata server with the REORG option

- run the optimizeIMDB command option of the metadata server script

- run the Metadata Analyze and Repair tools (except for the Metadata Server Cluster Synchronization tool, which runs on a server that has been started with clustering)

Note: It is recommended that you stop the other nodes in the cluster before performing one of these functions.

After you perform one of these functions, you must restart (or stop and start) the node to place it in cluster mode as the controller node. Then, start the other nodes in the cluster. The controller node updates the other nodes with the new data from the recovery, REORG, optimizeIMDB, or analyze and repair operation.

See Also

Viewing Logs for Clustered Metadata Servers

The logging facility creates a separate log for each node in a metadata server cluster. The logs are stored on their respective host machines. They are also available for viewing in SAS Management Console, as described in the next topic.

Monitoring Clustered Metadata Servers

Monitoring Clustered Metadata Servers in SAS Management Console

SAS Management Console enables you to view the overall status of a metadata server cluster and to individually monitor each node in the cluster.

To view the overall status of the cluster:

1. In the SAS Management Console **Plug-ins** tab, expand the **Metadata Manager** node. Right-click the **Active Server** node and select **Properties**.

   The Active Server Properties window appears.

2. Select the **Cluster** tab to see the overall status of the cluster (including the presence or absence of a quorum) and the status of each of the nodes in the cluster:

To view more detail about the individual nodes in a cluster:

1. Navigate to **Server Manager** ⇨ **SASMeta** ⇨ **SASMeta - Logical Metadata Server**.

2. Expand **SASMeta - Logical Metadata Server**. Each node appears on a separate line, as shown in the following example:
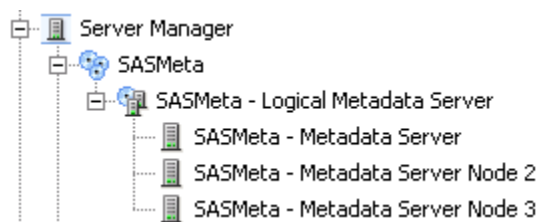
3. Select a node and connect to it. You can now do the following:

   • Use the tabs on the right pane to view the node's connections, clients, options, loggers, and log events.

   • Select **Stop** to stop only the selected node.

   • Select **Pause**, **Resume**, **Quiesce**, or **Validate**. These actions affect the entire cluster.

Monitoring Clustered Metadata Servers in SAS Environment Manager

Effective with SAS 9.4M2, SAS Environment Manager supports monitoring of SAS Metadata Server clusters. To view status indicators and metrics for the cluster:

1. On the **Resources** tab, select **Platforms**. In the list of platforms, select **SAS 9.4 Application Server Tier**.

   Deployment-wide information is displayed at the top of the page, including the message **Metadata Clustered: Yes**, as shown here:

2. Select **Monitor** and then select a time period to display.

3. Select **Indicators**, and then scroll down to display Metadata Cluster Nodes Available, Metadata Cluster Nodes Defined, Metadata Cluster Percent Available, and Metadata Cluster Quorum Available. For example, here is the Nodes Available indicator:
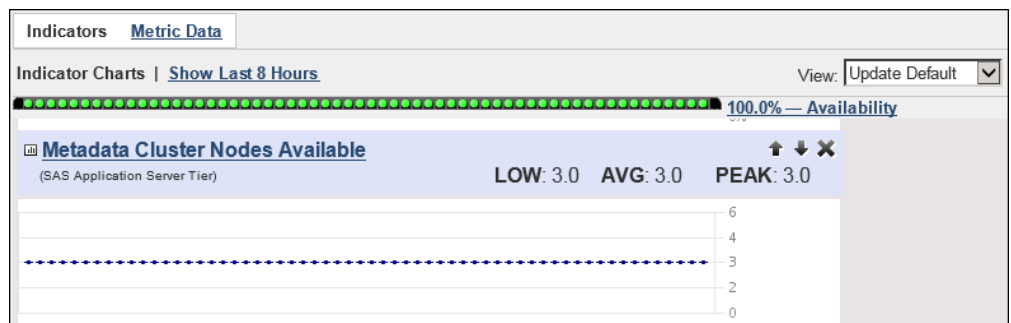
   To see the information in tabular format, click **Metric Data**.

SAS Environment Manager 2.4 includes the following predefined alerts for metadata server clusters:

- a low-severity alert for cluster availability that is less than 100%

- a medium-severity alert for a change of state in the cluster quorum

These alerts are activated by default if you initialize SAS Environment Manager Extended Monitoring.

For more details, see the *SAS Environment Manager: User's Guide* and the Help for SAS Environment Manager.

Recovering a Clustered Metadata Server

Using SAS Management Console to Recover

You cannot use the recovery facility in SAS Management Console to recover a metadata server that is operating in clustered mode. Instead, you must recover just one of the nodes in the cluster. When you restart the cluster after the recovery, the recovered data is automatically propagated to the other nodes.

Follow these steps:

1. Stop all of the nodes in the cluster. See "Stop a Clustered Metadata Server" on page 220.

2. Use the **-startNoCluster** option of the metadata server script to start one of the nodes without clustering. See "Start a Single Node without Clustering" on page 221.

3. Use SAS Management Console to connect to the node and perform the recovery. For details, see "Recovering the SAS Metadata Server" on page 178.

   CAUTION:
   Recovering configuration files from a backup is not recommended for clustered servers. Backed up configuration files could contain node-specific paths or options.

4. Restart (or stop and start) the node to place it in cluster mode as the controller node.

5. Start the other nodes in the cluster. The controller node updates the other nodes with the new data from the recovery operation.

Using the -recover Option

Starting with SAS 9.4 M1, you can also choose to use the **-recover** option of the metadata server script. This option starts a single node as the controller node and restores its metadata repository from the most recent backup.

The **-recover** option does not provide roll-forward recovery, recovery of configuration files, or recovery from a backup other than the most recent backup.

To use this option, follow these steps:

1. Stop all of the nodes in the cluster.

2. On one of the nodes, run the metadata server start-up script with the **-recover** option. The node is started as the controller node, and its metadata repository is restored from the most recent backup.

3. Start each of the other nodes by running the metadata server start-up script with the **-start** option. The controller node updates the other nodes with the new data from the recovery operation.

   Note: If a node does not start successfully, use the **-recover** option to start that node.

Migrating a Clustered Metadata Server

When migrating from a SAS 9.4 deployment that includes a clustered metadata server, run the SAS Migration Utility only on the initially configured metadata server. The migration utility migrates the server's metadata, but it does not migrate any cluster configuration information that is included in the metadata.

If you want the target deployment to use clustering, run the SAS Deployment Wizard on each additional node in the target deployment, as described in "Step 2: Install and Configure Additional Metadata Server Nodes" on page 217.

Part 6

Managing SAS Metadata

Chapter 17

Managing SAS Metadata Repositories

About SAS Metadata Repositories

A metadata repository is a physical location in which a collection of related metadata objects is stored. Metadata repositories are managed by a repository manager, which is a library of tables that define the set of repositories that can be accessed by a SAS Metadata Server. A metadata server cannot be started without a repository manager. Each metadata server can have only one repository manager.

The SAS Metadata Server supports three types of metadata repositories:

Foundation repositories
    are the required metadata stores for SAS Metadata Servers. Each metadata server has one foundation repository that is created by default when the metadata server is configured. The default name of the foundation repository is **Foundation**. (If you performed a custom installation, then you might have assigned a different name.) You cannot create more than one foundation repository.

Custom repositories
>are optional metadata stores for SAS Metadata Servers that can be configured in addition to the foundation repository. Custom repositories are useful for physically separating metadata for storage or security purposes.
>
>For example:
>
>- If your metadata storage requirements exceed the capacity of a single metadata storage location, you could create custom repositories on separate storage devices that are accessible to the metadata server.
>
>- If you have metadata that is highly sensitive, you could create a custom repository for that metadata and store it in a directory that has an exceptionally high level of protection from unauthorized access.
>
>Custom repositories appear as separate folders in the Folders view of SAS Management Console and in other client applications.
>
>To create a custom repository, see "Create a Custom Repository" on page 229.

Project repositories
>are metadata repositories that serve as isolated work areas or playpens. Project repositories are available for SAS Data Integration Studio only. In general, each user who participates in change management has his or her own project repository.
>
>A project repository enables a user to check out and lock metadata from a foundation repository or a custom repository so that the metadata can be modified and tested in a separate area. When the metadata is ready, it is checked in and unlocked so that it is available to other users.
>
>For information about creating project repositories, see "Setting Up Change Management" in *SAS Intelligence Platform: Desktop Application Administration Guide*.

The SAS Deployment Wizard creates just one repository, which is the foundation repository. If you create additional repositories, then the metadata in a given repository can be associated with metadata in other repositories as follows:

- Metadata in a custom repository can be associated with metadata in the foundation repository and with metadata in all other custom repositories.

- Metadata in the foundation repository or in any custom repository can be checked out to any project repository, as long as the user has the appropriate permissions.

- Metadata that is checked out to a project repository can be associated with metadata in the foundation repository or with metadata in any custom repository (but not with metadata that is checked out to a different project repository).

Note: An additional repository called BILineage is created automatically the first time an unrestricted administrative user logs on to SAS Management Console. This repository stores metadata lineage and impact analysis data that is generated by SAS Data Integration Studio, SAS OLAP Cube Studio, and the BI Lineage plug-in for SAS Management Console. The BILineage repository should not be used for any other purpose. For more information, see "Using the BI Lineage Plug-in" on page 250.

About SAS Metadata Repository Access Modes

The access mode of a metadata repository is the repository's intended access state. When the metadata server is online, the access state of a repository is equal to its access mode. When the metadata server is paused to an Offline state or an Administration state, the access state of a repository can change temporarily. See "How the Access State of the Metadata Server Affects the Access States of Metadata Repositories" on page 90. When the metadata server is resumed, the repository's access state returns to its registered access mode.

A repository's access mode is recorded in the Access attribute of the repository's metadata definition. When you create a new repository, you can place it in any of the following access modes:

ONLINE
> Clients can read, write, and update metadata. This access level is the default for new repositories.

READONLY
> Clients can read metadata but cannot write, update, or delete metadata.
>
> For example, if work on a project is complete, you can assign an access mode of READONLY to the project's repository so that users can continue to access the metadata without updating it.

ADMINISTRATION
> Only users who are in the **Metadata Server: Unrestricted** role can read or write metadata.
>
> For example, if metadata problems occur (for example, if the metadata in a repository becomes corrupted), you can assign an access mode of Administration while the issue is being analyzed and repaired.

OFFLINE
> Clients can neither read nor write metadata to the repository.
>
> For example, if work on a project is complete and users no longer need to access the metadata, you can assign an access mode of OFFLINE to free server memory. If the metadata is needed in the future, the access mode can be changed back to READONLY or ONLINE.

For information about changing a repository's access mode, see "Change the Properties of a Metadata Repository" on page 231.

Create a Custom Repository

To organize your metadata, the recommended practice is to create folders instead of creating separate repositories. See Chapter 18, "Working with SAS Folders," on page 243. However, you might want to create a custom repository in order to physically separate metadata for storage or security purposes.

When you create a custom repository, you should be aware that the metadata server must add, update, and delete repositories when no other activity is taking place on the server. Therefore, the server delays other client requests until the repository changes are complete. This delay might have a small effect on server performance.

To create a new custom repository, follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

2. On the **Plug-ins** tab, expand the **Metadata Manager** node.

3. Right-click the **Active Server** node, and select **New Repository**.

4. Follow the instructions on the wizard pages. As you answer the wizard's prompts, follow these guidelines:

   • Be sure to specify a unique name for the repository.

   • Generally, the path for a new repository should be *SAS-configuration-directory*/**Lev1/SASMeta/MetadataServer/MetadataRepositories**/*repository-name*. This is the default location for the initial foundation repository. By using this location, you will ensure that the directory inherits access permissions from the **MetadataServer** directory.

     If you choose to specify a different path, follow these guidelines:

     • Metadata repositories should be kept either on a local file system or a high-performance, high-availability network file system.

     • Do not place the directory under the **rposmgr** directory.

     • The directory should not contain any files other than the repository files.

     • Before completing the wizard, make sure that the process owner for the metadata server has credentials to create the directory (or to write to the directory, if it already exists).

     Note: If the last directory in the specified path does not exist, it is created. If the path includes any higher-level directories that do not exist, an error message is displayed.

   • If you do not want the repository to be ONLINE, you can select a different value in the Access attribute. See "About SAS Metadata Repository Access Modes" on page 229.

When you create a custom repository, SAS Management Console automatically does the following:

• creates a corresponding folder that appears in the **Folders** tab of SAS Management Console and in other client applications.

• creates a default access control template (ACT) that you can use to restrict user access to the repository. For details about repository access controls, see "Metadata Authorization Model" in *SAS Intelligence Platform: Security Administration Guide*.

You can also use these methods to create a custom repository:

• Use SAS language elements. If you use this method, then you must use SAS Management Console to initialize the repository. The initialization process creates the Default ACT and the resource templates.

• Use the sas-create-repository batch tool. See "Create a Repository (sas-create-repository)" on page 379.

Change the Properties of a Metadata Repository

You can change the name or the registered access mode of a metadata repository. For foundation repositories, you can also change the location of users' folders. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

2. On the **Plug-ins** tab, expand the **Metadata Manager** node and the **Active Server** node.

3. In the Active Server tree, right-click the repository whose properties you want to change, and select **Properties**.

4. If you want to change the repository's name, enter the new name on the **General** tab and click **OK**. When you rename a custom repository, the corresponding folder in the **Folders** tab is also renamed.

   CAUTION:
   Do not rename a custom repository if the repository (or one of its subfolders) contains an object that has an association with another object. Doing so could cause the associated objects to malfunction.

   When you rename a custom repository, the corresponding folder in the SAS Folders tree is automatically renamed to the new name.

5. If you want to change the repository's access mode, click the **Registration** tab and make a selection in the **Access** field. (For a description of the access modes, see "About SAS Metadata Repository Access Modes" on page 229.). Then click **OK**.

6. The **Registration** tab provides a field in which you can change the location of users' home folders. If you need to make a change to this field, it is recommended that you first contact SAS Technical Support for additional steps that might be required.

Copying, Promoting, Importing, Exporting, and Analyzing Your Metadata

Tools for Copying, Promoting, Importing, Exporting, and Analyzing Metadata

The SAS Intelligence Platform includes several tools to assist you in copying, promoting, importing, exporting, and analyzing your metadata. The following table lists each tool, the purpose of each tool, the tool's location in SAS Management Console (if applicable), and a reference for documentation about the tool:

Table 17.1 *Tools for Managing Metadata Repositories*

Tool Name	Purpose	Location in SAS Management Console	Documentation Reference
Metadata server backup and recovery facility	Copy all of your metadata repositories, the repository manager, and the contents of the metadata server's configuration directory to or from the configured backup location.	**Metadata Manager ⇨ Metadata Utilities ⇨ Server Backup**	Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169
Import Metadata and Export Metadata*	Copy metadata for relational data and DBMSs to and from external sources.	**Metadata Manager ⇨ Active Server ⇨ Import Metadata** and **Metadata Manager ⇨ Active Server ⇨ Export Metadata**	"About the Metadata Manager Import and Export Wizards" on page 233
Export SAS Package and Import SAS Package**	Promote individual objects or groups of objects from one location to another, either on the same metadata server or a different metadata server. You can also include associated physical content in the promotion.	In the **Folders** tab, right-click a folder and select **Export SAS Package** or **Import SAS Package**.	Chapter 20, "Promotion Tools Overview," on page 265 and SAS Management Console Help
Copy, Paste, and Paste Special	Copy and paste a group of objects or folders into a target folder on the same metadata server.	In the **Folders** tab, right-click a folder and select **Copy**. Then right-click another folder and select **Paste** or **Paste Special**.	SAS Management Console Help
SAS Migration Utility	Populate a SAS deployment with metadata, associated physical data, and configuration information from an existing deployment.	Not applicable. The utility is executed from a command line.	"SAS Migration Utility Reference" in *SAS Intelligence Platform: Migration Guide*
Metadata Analyze and Repair	Run selected tests on metadata to locate common problems, and (when possible) repair problems that are found.	**Metadata Manager ⇨ Active Server ⇨ Analyze/ Repair Metadata**	"Analyzing and Repairing Metadata" on page 236

* SAS Data Integration Studio enables you to perform this function with change analysis. The change analysis feature enables you to compare imported metadata to existing metadata in the repository and selectively apply changes. For details, see the product Help for SAS Data Integration Studio.

** This feature is also available in SAS Data Integration Studio and SAS OLAP Cube Studio.

About the Metadata Manager Import and Export Wizards

The Metadata Manager Import and Export Wizards enable you to copy metadata for relational data and DBMSs to and from external sources. To use this feature, you must license and install the appropriate SAS Metadata Bridge. For information about SAS Metadata Bridges, contact your SAS representative.

You can import and export relational metadata in any format that is accessible with a SAS Metadata Bridge. Relational metadata includes the metadata for the following objects: tables, columns, indexes, and keys (including primary keys and foreign keys). This feature is useful when you have created a data model using other data warehousing software, and you want to import it for use with SAS; or when you have created a data model in SAS, and you want to export it for use with other data warehousing software.

To export metadata, expand the **Metadata Manager** node in SAS Management Console, right-click the **Active Server** node, and select **Export Metadata**. In the Metadata Export Wizard, you specify the export format, navigate to the folders that contain the table objects to be exported, and select the tables. You then specify the path and file name to which the metadata is to be exported.

To import metadata, expand the **Metadata Manager** node in SAS Management Console, right-click the **Active Server** node, and select **Import Metadata**. In the Metadata Export Wizard, you specify the import format, specify the path and file name that contains the metadata to be imported, and specify the library to which the metadata is to be imported.

For more information about using the wizards, see the Help for SAS Management Console. Error messages generated by the wizards are written to SASMCErrorLog.txt, which is located in one of the following paths:

- On Windows Vista, Windows 7, and Windows Server 2008 systems: `C:\Users\`*`user`*`\AppData\Roaming\SAS\SASManagementConsole\9.4`

- On other Windows systems: `C:\Documents and Settings\`*`user`*`\Application Data\SAS\SASManagementConsole\9.4`

- On UNIX systems: *`user's-home-directory/`*`SAS/SASManagementConsole/9.4`

Note: SAS Data Integration Studio provides enhanced Import and Export wizards to copy metadata for relational data and DBMSs to and from external sources. These wizards provide a change analysis capability that enables you to compare the imported metadata to existing metadata in the repository and selectively apply changes. For details, see the product Help for SAS Data Integration Studio.

Moving a Metadata Repository to a New Location on the Same Metadata Server

In most cases, it is neither necessary nor advisable to move metadata repositories to a new location, for the following reasons:

- Metadata server performance is unlikely to be improved by moving repositories to a faster file system. The metadata server uses an in-memory database, and file system updates are performed in a background thread in SAS 9.2 and later releases.

- Repositories are unlikely to require a larger amount of disk space than is available on the current file system. In SAS 9.2 and later releases, repositories are stored in compressed data sets. Except in the very largest deployments, storage requirements are unlikely to exceed a total of 1 gigabyte.

  Note: Metadata server logs can cause disk space to fill up if additional logging has been enabled for troubleshooting or other purposes. To avoid this situation, be sure to reset logging to the default level when troubleshooting has been completed.

Because of the complexities involved, metadata repositories should not be moved unless there is a strong reason to do so. Implementation of high-availability failover for the metadata server is one of the few situations in which it might be appropriate to consider moving metadata repositories to a network-available file system.

CAUTION:
Moving a metadata repository is a complex task and is not necessary in most cases. If you determine that you need to move a metadata repository (for example, to support failover), it is strongly recommended that you first contact SAS Technical Support for advice and assistance.

Registering a Metadata Repository

When you register a metadata repository, information about the repository is added to the repository manager so that the repository can be accessed by clients.

You might need to register a repository if you have moved the repository to a new location on the same metadata server and you want to register its new location.

To register a repository:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

2. In the Metadata Manager, right-click the **Active Server** node and select **Register Repository**. Then follow the instructions that are provided by the wizard.

Unregistering and Re-registering a Metadata Repository

When you unregister a metadata repository, information about the repository is removed from the repository manager, and the repository can no longer be accessed by clients.

You might need to temporarily unregister a repository when you are moving the repository to a new location on the same metadata server.

CAUTION:
Unregistering the foundation repository makes the metadata server unusable. Therefore, you should unregister a foundation repository only in special situations such as moving the repository.

The following rules apply to unregistering a repository:

- You cannot unregister a foundation repository if other repositories (including custom repositories or project repositories) are registered. You must first unregister the other repositories.

- Unregistering a foundation repository requires special credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

- You cannot unregister a repository that is in an OFFLINE or READONLY state. You must first change the access mode of the repository to ONLINE or ADMINISTRATION. See "Change the Properties of a Metadata Repository" on page 231.

To unregister a repository:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. In the Metadata Manager, expand the **Active Server** node.

3. Right-click the repository that you want to unregister, and select **Unregister Repository**.

To re-register a repository:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

2. In the Metadata Manager, expand the **Active Server** node.

3. Right-click the **Active Server** node, and select **Register Repository**. Then follow the instructions on the wizard pages.

   Note: When a repository is created, the repository type (foundation, custom, or project) is recorded in the repository. When you re-register a repository, the repository type must match the type that was previously recorded.

Deleting a Metadata Repository

When you delete a metadata repository, all of the repository's metadata and metadata containers are deleted. In addition, the repository's registration is removed from the repository manager.

You might want to delete a repository from its previous location if you have moved the repository to a new location on the same metadata server.

CAUTION:
Deleting the foundation repository makes the metadata server unusable.
Therefore, you should delete a foundation repository only in special situations, such as those that are described in the preceding list.

The following rules apply to deleting a repository:

- You cannot delete a foundation repository if other repositories (including custom repositories or project repositories) are registered. You must first unregister the other repositories.

- Deleting a foundation repository requires special credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

- You cannot delete a repository that is in an OFFLINE or READONLY state. You must first change the access mode of the repository to ONLINE or ADMINISTRATION. See "Change the Properties of a Metadata Repository" on page 231.

To delete a repository:

1. Log on to SAS Management Console as a user with the appropriate credentials.

2. In the Metadata Manager, expand the **Active Server** node.

3. Right-click the repository that you want to delete, and select **Delete**.

You can also use the sas-delete-repository batch tool. See "Delete a Repository (sas-delete-repository)" on page 380.

Analyzing and Repairing Metadata

About the Metadata Analyze and Repair Tools

Note: If you are running a non-clustered metadata server, the Metadata Analyze and Repair Tools can be run without stopping any SAS servers. For clustered metadata servers, the only Metadata Analyze and Repair tool that can run while in quorum is "Cluster Synchronization". The other Metadata Analyze and Repair tools for clustered metadata servers must be run on a single node that has been started without clustering (-STARTNOCLUSTER option).

The Metadata Analyze and Repair tools, which can be accessed from the **Metadata Manager** node in SAS Management Console, enable you to run selected tests on metadata to locate common problems. When possible, the tools also repair problems that the analysis has identified.

The Analyze and Repair tools provide the following options:

- You can analyze one repository at a time, or you can analyze multiple repositories simultaneously.

- You can choose to run the repair phase immediately following the analysis, or you can examine the analysis results and then apply the repairs in a separate step. (The latter option is recommended.)

- Several metadata analysis and repair tools are available, as described in the following topic. You can run either a single tool or multiple tools in a single execution.

- Analysis and repair of metadata should be performed whenever it is recommended in the metadata server log or when other server maintenance tasks are performed. It might be appropriate to repair metadata on a quarterly basis for sites with significant metadata updates. You might want to adjust the frequency needed for metadata repairs for your site based on the results of the analysis.

When you run the Metadata Analyze and Repair tools, the metadata server is automatically paused to ADMINISTRATION mode. Therefore, you might want to run the tools during times of low metadata server activity. The ADMINISTRATION state prevents metadata changes from occurring while the analysis process is running, except that unrestricted users can continue to change metadata during this time. The server is automatically resumed when the analysis and repair processes is completed.

Note: For clustered metadata servers, in most cases you should run the Metadata Analyze and Repair tools on a single node that has been started without clustering.

An exception is the Metadata Server Cluster Synchronization tool, which works only on a server that has been started with clustering.

See Also

To use these tools in batch mode, see "Analyze and Repair Metadata (sas-analyze-metadata)" on page 382.

Metadata Analyze and Repair Tool Descriptions

Verify Metadata Files

The **Verify Metadata Files** tool analyzes key metadata server files to determine whether they are corrupted and, when possible, recommends repairs that can be applied. This tool might be useful if a metadata server failure occurs or if a user experiences unexpected results when accessing the server. For example, it might be useful if a repository cannot be opened, a new repository cannot be registered, or metadata that was previously accessible cannot be accessed. It might also be useful if metadata associations have been broken.

Verify Associations

The **Verify Associations** tool checks the metadata repository for associations in which one or the other associated object does not exist. This tool might be useful if a user experiences unexpected results when accessing the server.

When you run this tool, do not select the option to **Repair immediately**. When the analysis is finished, select **View log**. The log provides the following information:

- The total number of associations in the repository.

- The number of broken associations in the repository. These are associations within the same repository that involve at least one metadata object that does not exist. Broken associations are candidates for removal by the repair tool.

   To see details about each broken association, access the metadata server log and search for the string `AnalyzeAssociations`. You can then remove the broken associations by proceeding with the repair step.

- The number of suspect associations in the repository. These are associations between objects in different repositories that involve at least one metadata object that cannot be located. The Verify Associations tool cannot determine whether the missing object is actually missing or whether it resides in a repository that is currently offline. Therefore, suspect associations cannot be repaired.

Metadata Server Cluster Synchronization

The **Metadata Server Cluster Synchronization** tool, which is new with SAS 9.4M3, verifies that metadata is synchronized among all the nodes of a metadata server cluster. If metadata differences are found, run the tool with the repair option, which does the following:

- sets a flag on the server indicating that the metadata is out of sync

- writes messages about the issue to the Analyze and Repair log file

- generates an alert email notifying the administrator that a node is out of sync and needs to be stopped and restarted

The next time the problem node is restarted, the flag causes the node's metadata to be synchronized with metadata on the controller node.

Note: Metadata Server Cluster Synchronization works only on a metadata server that has been started with the clustering option. No analysis is performed if this tool is run on a non-clustered metadata server or on a clustered server that contains only one node.

Verify Permissions

The **Verify Permissions** tool verifies that permission objects exist only in the Foundation repository and recommends repairs that can be applied.

Verify Authentication Domains

The **Verify Authentication Domains** tool checks authentication domain objects to ensure that the object names are valid and unique. It also checks to ensure the valid usage of these objects. When possible, the tool recommends repairs that can be applied.

Orphaned Objects

The orphaned objects tools locate metadata objects that are no longer being referenced. Metadata can become orphaned when an application deletes a metadata object but does not delete metadata that is associated with the deleted object. Orphaned objects are candidates for removal by the repair tool.

It can be helpful to run the orphaned metadata tools after migrating or promoting your metadata from a previous SAS release. In particular, deleting orphaned objects after upgrading from SAS 9.1.3 can free up a significant amount of disk space. After you run one of these tools with the repair option, be sure to back up the metadata server with the Reorganize Repositories option to reclaim the unused disk space.

You can choose from the following orphaned metadata tools:

*   **Orphaned Access Controls**
*   **Orphaned Directories**
*   **Orphaned Job Components**
*   **Orphaned Directories**
*   **Orphaned Properties**
*   **Orphaned Source Code**

Validate SAS Folders

The **Validate SAS Folders** tool analyzes the integrity of objects contained in the SAS Folders tree and, when possible, recommends repairs that can be applied. This tool identifies issues such as duplicate, missing, or invalid object names; objects that are present in more than one folder; and objects that are stored in a different repository than their parent folders.

Running the Metadata Analyze and Repair Tools

Note: If you are running a non-clustered metadata server, the Metadata Analyze and Repair Tools can be run without stopping any SAS servers. For clustered metadata servers, the only Metadata Analyze and Repair tool that can run while in quorum is "Cluster Synchronization". The other Metadata Analyze and Repair tools for clustered metadata servers must be run on a single node that has been started without clustering (-STARTNOCLUSTER option).

CAUTION:

Be sure to back up the metadata server before you perform the repair phase of the Metadata Analyze and Repair Tool. See "Using SAS Management Console to Run an Ad Hoc Metadata Server Backup" on page 176.

To run the Metadata Analyze and Repair Tool:

1. If you have configured a clustered metadata server, stop all of the nodes in the cluster. Then start a single node without clustering. See "Start a Single Node without Clustering" on page 221.

    Note: Omit this step if you will be selecting the Metadata Server Cluster Synchronization tool, which works only on a server that has been started with clustering.

2. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

3. Expand the **Environment Management** and **Metadata Manager** nodes. Then right-click the **Active Server** node, and select **Analyze/Repair Metadata**. (You can also start the process by right-clicking one of the repositories that appear under the **Active Server** node.)

4. The following message is displayed: `The metadata server will be paused in administration mode to run the analyze and repair tools. Only administrative users will be able to access the metadata server while the process is running. Do you wish to proceed?` Click **Yes** if you are ready to begin the analysis. The server will be paused after you complete the next two wizard pages.

5. On the first wizard page, select one or more repositories to analyze and repair. Click **Next**.

6. The next wizard page lists the analysis tools that are available:
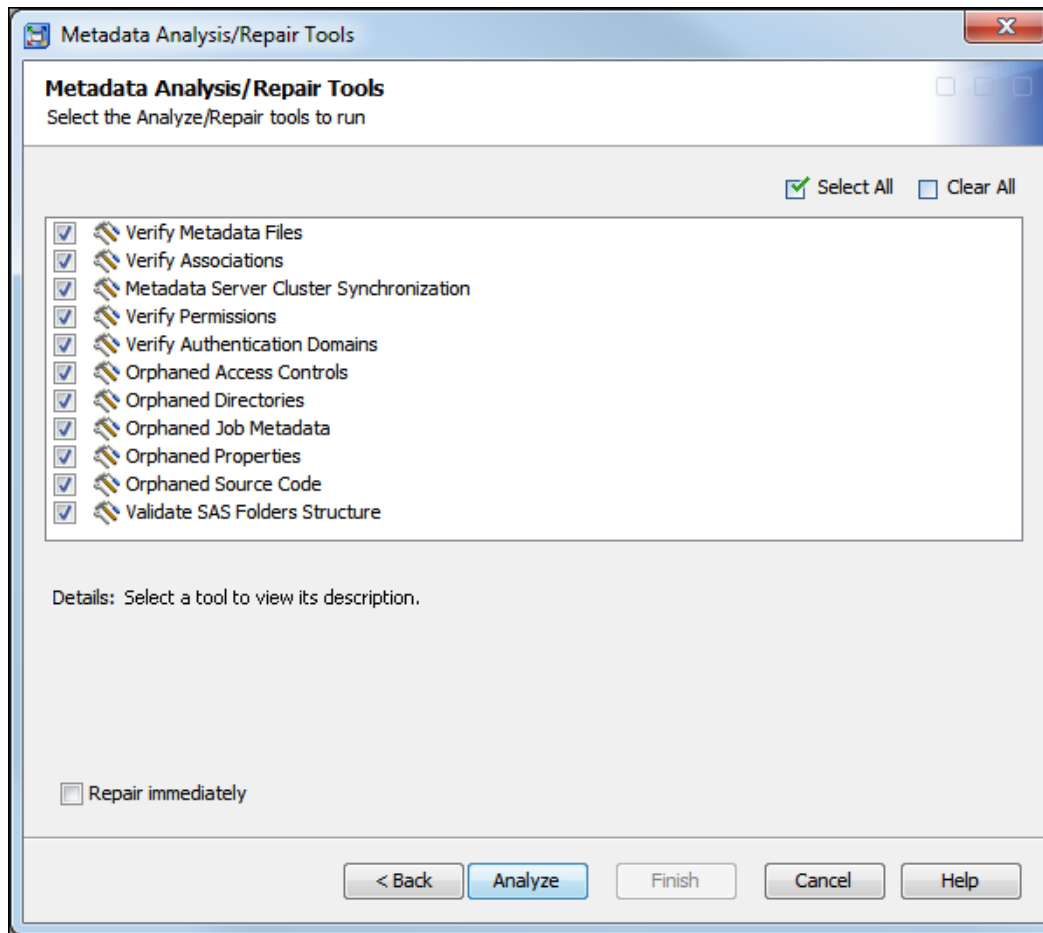
Select or deselect the check boxes to specify the analyses to perform. For descriptions of each tool, see "Metadata Analyze and Repair Tool Descriptions" on page 237.

You can also click the check box to **Repair immediately**, but it is usually recommended that you perform the repairs in a separate step.

Click **Analyze**.

Note: If you select **Repair immediately**, the **Analyze** button is replaced by the **Repair** button. You can then click this button and proceed as described in steps 9 and 10.

7. A message is displayed stating that the server is being paused to Administration mode. The analysis is then performed. When it is finished, the results are displayed as shown in the following example:
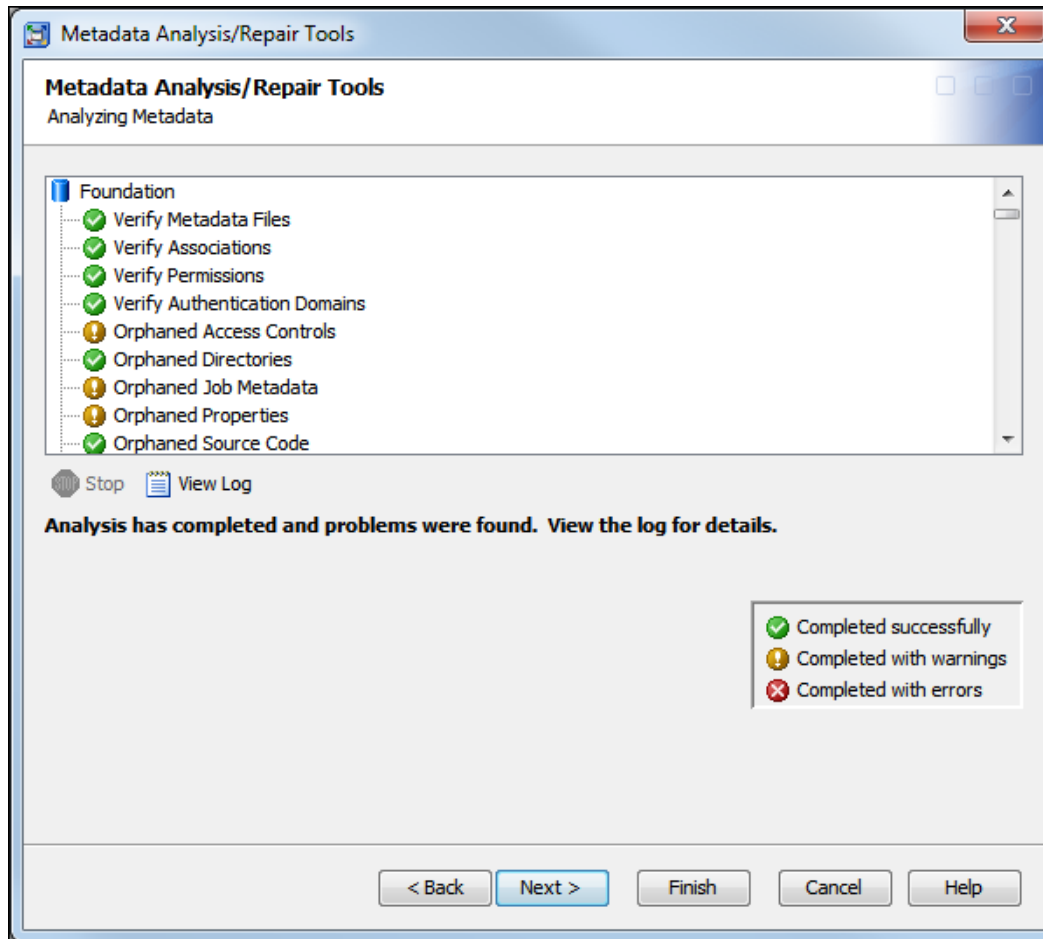
8. If no problems are found, the following message is displayed: `Analysis has completed with no problems found.` You can then click **Finish** to exit the wizard.

   If problems are found, the following message is displayed: `Analysis has completed and problems were found. View the log for details.` Click **View Log** to see information about the errors. Additional details might also be available in the metadata server log.

9. The next wizard page displays a list of the analysis tools that found problem situations. Select one or more tools to run in repair mode, and click **Repair**.

10. A message reminds you to back up your metadata before running the repairs. Click **Yes** to continue; or if you have not taken a backup, click **No**, run the backup, and then restart the Analyze and Repair process from the beginning.

11. If you click **Yes**, the repairs are executed. A window indicates whether each repair was completed successfully.

    You can click **View Log** to see details about the repairs. Then click **Finish** to exit the wizard.

12. If you ran the tool on a clustered metadata server, take an adhoc metadata backup of the single node that is running after repairing the metadata. You do not need to run the tools on the other nodes. Then, restart (or stop and start) the single node to place it in cluster mode as the controller node. Finally, start the other nodes in the cluster with the recover option to refresh their repository from the recent backup.

Note: Omit this step if you selected the Metadata Server Cluster Synchronization tool while running in clustered mode..

Chapter 18

Working with SAS Folders

About the SAS Folder Structure

Purpose of the SAS Folder Structure

SAS Intelligence Platform clients use a hierarchy of SAS folders to store metadata for the following:

- content such as channels, cubes, data explorations, generated transformations, information maps, jobs, libraries, mining results, OLAP schemas, prompts, reports, stored processes, subscribers, and tables

- SAS system content that is not directly accessed by business users

When you install SAS, a set of default SAS folders is created. The folders are arranged in a structure that segregates system information from business information, provides personal folders for individual users, and provides an area for shared data. Within this overall structure, you can create a customized folder structure that meets the information management, data sharing, and security requirements of your organization.

If you have installed web applications such as SAS Web Report Studio, then some SAS folders (for example, folders that contain metadata for reports) have associated physical content that resides on the SAS Content Server. In these instances, a procedure called

content mapping maps the metadata folder structure to corresponding physical folders that have the same organization. Content mapping is automatically configured when you install your system.

All SAS folders that the user has permission to view are displayed in the **Folders** tab of SAS Management Console and in the Folders view of the **Administration** tab in SAS Environment Manager. Most other client applications display SAS folders only if they contain content that is relevant to the application, subject to the user's permissions.

Note: In SAS Data Integration Studio, metadata objects that are created in project repositories are not displayed in the **Folders** tab until they are checked in.

About the Initial Folder Structure for New Installations

For a newly installed SAS 9.4 system, the initial folder structure is similar to the following:
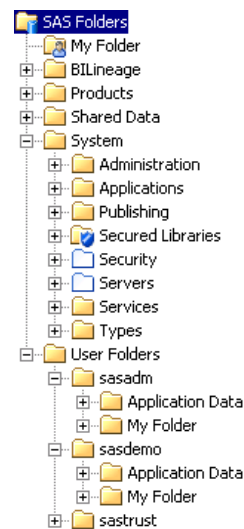
CAUTION:
> **Renaming, moving, or deleting SAS folders and the objects that they contain can cause unpredictable results.** Before renaming, moving, or deleting an object or a folder, see the guidelines in "Best Practices for Managing SAS Folders" on page 247 and "Best Practices for Maintaining Associations among Objects in SAS Folders" on page 248.

The initial folder structure includes the following main components:

SAS Folders
   is the root folder for the folder structure. This folder cannot be renamed, moved, or deleted. It can contain other folders, but it cannot contain individual objects.

My Folder ( )
   is a shortcut to the personal folder of the user who is currently logged on. For details, see the other instance of "My Folder" in this list.

BILineage
   is the root folder for the BILineage metadata repository. This repository stores results from scans that have been run using the BI Lineage plug-in. This folder should not be renamed, moved, or deleted. The repository and folder should not be used for any

purpose other than storing scan results. For more information about the BI Lineage plug-in, see "Using the BI Lineage Plug-in" on page 250.

Products
> contains folders for individual SAS products. These folders contain content that is installed along with the product. For example, some products have a set of initial jobs, transformations, stored processes, or reports which users can modify for their own purposes. Other products include sample content (for example, sample stored processes) to demonstrate product capabilities.
>
> Where applicable, the content is stored under the product's folder in subfolders that indicate the release number for the product.
>
> *Note:* During installation, the SAS Deployment Wizard allows the installer to assign a different name to this folder. Therefore, your **Products** folder might have a different name.

Shared Data
> is provided for you to store user-created content that is shared among multiple users. Under this folder, you can create any number of subfolders, each with the appropriate permissions, to further organize this content.
>
> *Note:* You can also create additional folders under **SAS Folders** in which to store shared content.

System
> contains SAS system objects that are not directly accessed by business users. This folder contains the following folders:
>
> **Administration**
>> is not currently used.
>
> **Applications**
>> contains folders for individual SAS applications that have system objects. Under these folders, the objects are stored in subfolders that correspond to individual release numbers.
>
> **Publishing**
>> contains channel and subscriber objects that are used by the Publishing Framework.
>
> **Secured Libraries**
>> contains secured data folders, secured library objects, and secured table objects that have been created to support metadata-bound libraries. See the *SAS Guide to Metadata-Bound Libraries*.
>
> **Security** and **Servers**
>> contain references to security objects (users, user groups, roles, access control templates, and authentication domains) and server objects. The white folder icons ( ) indicate that these are virtual folders. The folders are displayed only in SAS Management Console to support operations such as promotion. See "Promoting Security Objects and Server Objects" on page 297.
>
> **Services**
>> is used by SAS BI Web Services to store metadata for generated web services.
>
> **Types**
>> contains type definitions for public objects that exist on this metadata server.

User Folders
contains folders that belong to individual users. These folders are referred to as the users' home folders. The name of each home folder is based on the value of the user's Name field in the User Manager plug-in for SAS Management Console.

The first time a user logs on to an application that requires a home folder, the user's home folder is automatically created. That same folder is then used by other applications that the user logs on to.

As a default, users do not have Write access to their home folders. However, each home folder contains a folder called "My Folder" (referred to as the user's personal folder) to which the user has full access.

Note: The following details apply to the creation of users' home folders:

- When home folder names are created, any slash (/) or back-slash (\) characters in the user's name are converted to hyphens. For example, if the user's Name field contains **Jane/Doe**, then **Jane-Doe** is used as the folder name. Leading and trailing spaces, as well as unprintable characters, are also removed.

- In a custom installation, the SAS Deployment Wizard allows the installer to choose whether to create a **User Folders** folder. Therefore, your site might not have a **User Folders** folder. If you do not have a **User Folders** folder, but you would like to create one, see "Change the Properties of a Metadata Repository" on page 231.

Each home folder contains the following folders:

Application Data
is a standard location in which applications can store user preferences and other user-specific configuration information. The folder is created under the user's home folder the first time an application needs to use it. The folder is to be accessed only by applications and is not accessed directly by the user.

My Folder
is the user's personal folder. This folder exists under each user's home folder. It is the standard location for storing content (for example, information maps and reports) that the user creates, similar to the **My Documents** folder in Microsoft Windows.

The default security settings for **My Folder** are as follows:

- The folder is visible only to the owning user and to unrestricted users.

- The owning user has all permissions on this folder except WriteMetadata. Only the owning user and unrestricted users have permission to add content to or remove content from the folder. However, the owning user cannot rename, move, delete, or change permissions on the folder.

When the owning user is logged on, a pointer to the user's personal folder (**My Folder**) appears at the top of the folder hierarchy.

If you have defined one or more custom repositories, then the folder structure also includes a top-level folder that corresponds to each custom repository. For more information about custom repositories, see "Create a Custom Repository" on page 229.

Project repositories do not appear in the **Folders** tab of SAS Management Console. They appear only in SAS Data Integration Studio.

Using Search Folders and Favorites Folders

Search folders are represented by the following icon: 🔍. The **Search** tab in SAS Management Console enables users to search for metadata objects based on object names, locations, descriptions, object types, creation or modification dates, keywords, and responsibilities. After performing a search, click the **Save** button on the **Search** tab to specify a folder and location in which to store the search criteria. Opening a search folder causes the search to be rerun and updated search results to appear.

Favorites folders are represented by the following icon: ⭐. Favorites folders enable users to store references to often-used metadata objects. To create a Favorites folder, right-click a folder in the SAS Folders tree (for example, your personal folder), select **New** ⇨ **Favorites Folder**, and specify a folder name. Users can further organize favorites by selecting **New Favorites Group** when a Favorites folder is selected. To store a favorite, right-click the Favorites folder (or Favorites group), select **Add Favorite**, and navigate to the object that you want to reference.

Best Practices for Managing SAS Folders

To ensure the integrity of your system, follow these best practices when interacting with SAS folders in SAS Management Console:

Use personal folders for personal content and use shared folders for content that multiple users need to view.

By default, users cannot view other users' personal folders. Therefore, personal folders should be used for content that needs to be viewed and used only by the owning user. If the content needs to be viewed or used by other users, then it should be placed under the **Shared Data** folder or in a new folder structure that you create under **SAS Folders**.

To ensure secure and efficient sharing of content, the system administrator should create a folder structure for shared data that meets the needs of the organization. The appropriate permissions can then be assigned to each folder. For guidance, see "Customizing Your Folder Structure" on page 255.

Use folders, instead of custom repositories, to organize content.

In most cases, folders are the preferred method for organizing content. Custom repositories should be created only when there is an overriding reason to physically segregate repository data sets. For more information, see Chapter 17, "Managing SAS Metadata Repositories," on page 227.

It is recommended that you not delete or rename the **User Folders** folder, even if you have permission to do so.

If you have a reason to delete or rename the **User Folders** folder, then you must change the metadata repository configuration to reflect the change. See "Change the Properties of a Metadata Repository" on page 231.

Do not delete or rename the home folder or personal folder (**My Folder**) of an active user, even if you have permission to do so.

As a best practice, do not rename an active user's home folder or personal folder. If you do so, a new (empty) personal folder will be created the next time the user

refreshes or logs on to an application that requires the folder. In addition, the contents of the renamed folder will not be visible to the user.

If you delete an active user's home folder or personal folder, the user will lose any existing personal content, and a new (empty) personal folder will be created the next time the user refreshes or logs on to an application that requires the folder.

Do not delete or rename the **Products** or **System** folders or their subfolders, even if you have permission to do so.

Deleting or renaming the **Products** or **System** folders or their subfolders could cause erroneous or unexpected behavior in client applications or solutions.

Use caution when renaming the **Shared Data** folder.

Renaming the **Shared Data** folder can affect associations and references to objects that are stored in this folder or its subfolders. For details, see "Best Practices for Maintaining Associations among Objects in SAS Folders" on page 248.

When you create new folders, the security administrator should set permissions to ensure appropriate access.

For details, see "Access Management" in *SAS Intelligence Platform: Security Administration Guide*.

Best Practices for Maintaining Associations among Objects in SAS Folders

To ensure that associations among objects are maintained, follow these best practices when interacting with folders and with objects that are stored in SAS folders.

Use available tools to identify associations among objects.

To identify objects that are associated with information maps and reports, you can use the BI Lineage plug-in for SAS Management Console. For details, see "Using the BI Lineage Plug-in" on page 250.

Effective with SAS 9.4M1, you can use the batch relationship reporting tools to identify relationships among a larger number of content items. For details, see Chapter 26, "Using the Batch Relationship Reporting Tools," on page 347.

To determine associations and dependencies, you can also right-click the object and select **Export SAS Package**. On the second wizard page, highlight the object, and then use the **Dependencies** and **Used by** tabs to see the object's associations. Click **Cancel** to exit the wizard.

Use the appropriate SAS applications to interact with metadata objects.

To interact with objects that are stored in SAS folders, users should use the specific SAS application that creates and manages those objects. For example:

- Use SAS Data Integration Studio to interact with jobs.

- Use SAS Information Map Studio to interact with information maps.

- Use SAS Web Report Studio to interact with reports.

Certain administrative tasks are exceptions. For example, administrators can use the **Folders** tab of SAS Management Console to perform the following tasks:

- manage permissions on folders (and, if necessary, on individual objects)

- use the Export SAS Package and Import SAS Package wizards to promote objects or groups of objects (and their associated physical data) from one metadata server to another

- use the Copy and Paste Special functions to copy objects or groups of objects (and their associated physical data) from one folder to another

Use caution when renaming folders and when moving objects from one folder to another.

Renaming folders, or moving objects from one folder to another, can affect associations and references to objects that are contained in a folder or its subfolders, as follows:

- Some types of objects are referenced using folder pathnames. Associations to these types of objects can break if you rename a folder that is part of the object's path or if you move the object to a different folder. For example:

  - Reports use folder paths to locate information maps. If the location of an information map changes, then you might need to edit associated reports to point to the new information map location.

  - Information maps and reports use folder paths to locate objects such as cubes, prompts, and stored processes. If the location of one of these objects changes, then you might need to edit associated reports and information maps to point to the new location. In SAS Information Map Studio, you can use the Resource Replacement feature to change the location for some types of objects.

  - Some prompts (including library prompts, data source prompts, and prompts that use dynamically generated data values) use folder paths to locate libraries, tables, and information maps. If the location of a library, table, or information map changes, you might need to edit the affected prompts to point to the new location.

- If you move a table or a library, or if you rename a folder that is in the path for a library or table, most associations to the library or folder will continue to work. Client applications will still be able to locate the library or table. Jobs, information maps, and reports that use the library or table will continue to function. The only exceptions are some types of prompts, as described in the preceding paragraph.

- Renaming folders and moving objects can affect the following SAS Information Delivery Portal content:

  - information maps that are displayed in information map viewer portlets

  - links to detailed data (for example, reports or stored processes) that are coded in portlets

  - report references that have been published to WebDAV

Use caution when renaming objects in SAS Management Console.

If you rename an object that is associated with other objects, then associated objects can malfunction. For example, if you rename a library or table that is used by an information map, then you will need to use the Resource Replacement feature of SAS Information Map Studio to specify the new library or table name.

Note: For information about renaming objects and folders in client applications other than SAS Management Console, see the documentation for that application.

If you need to move content from one folder location to another (for example, from a user's personal folder to a shared folder), then use either Export SAS Package and Import SAS Package or Copy and Paste Special. Content movement should be carefully planned and executed so that metadata associations are maintained.

Correct use of **Export SAS Package** and **Import SAS Package** or **Copy** and **Paste Special** will maintain associations among objects as well as relationships between metadata and physical data.

For example, if you use one of these methods to move an information map that is associated with a report, then the association will be maintained. If you move the information map using a different procedure, then the report will no longer work.

For instructions for using **Export SAS Package** and **Import SAS Package**, see Chapter 20, "Promotion Tools Overview," on page 265.

Using the BI Lineage Plug-in

About the BI Lineage Plug-in

The BI Lineage plug-in for SAS Management Console identifies connections between BI objects. To generate lineage information, you use the BI Lineage plug-in to run a scan on a subset of folders in the SAS Folders tree. The scan examines reports and information maps that are stored in the selected folders, and identifies other objects (regardless of their location) that are connected to those reports and information maps. The starting point for the scan can be any folder (other than the root folder) in the SAS Folders tree.

Scan results are stored in a tree structure that can be viewed on the **Plug-ins** tab, as shown in the following example:
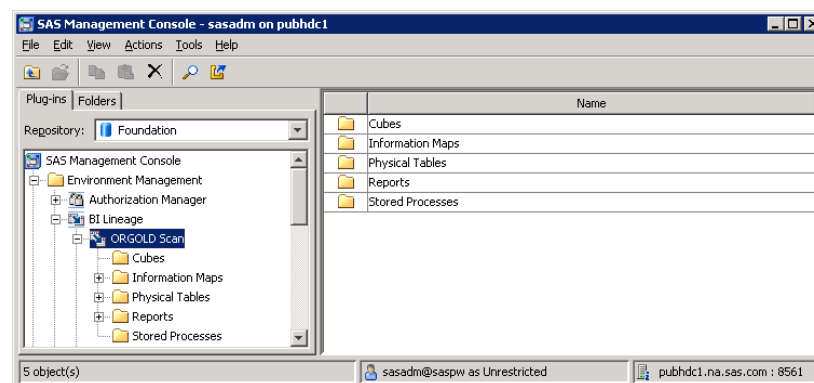
Authorized users can navigate the tree for a particular scan, select a BI object that was included in the scan, and display the object's lineage or reverse lineage report. Here is an example of a lineage report for a report object, showing an information map and two tables that are connected to the report.
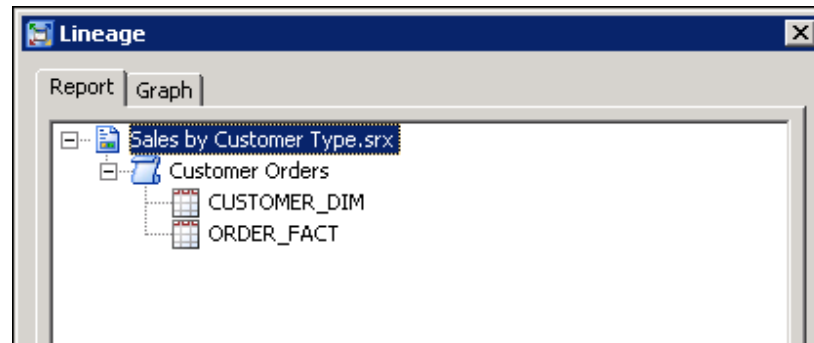
Information can also be displayed in graphical form, as shown here:
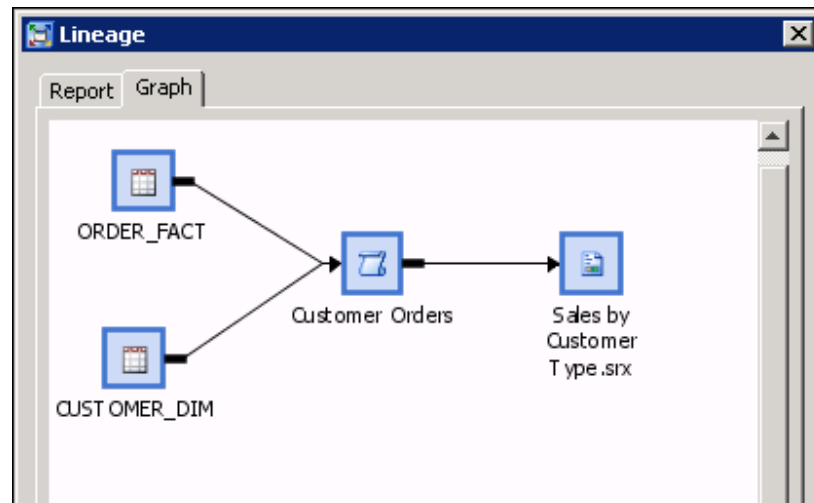
For details about information that is available in scan results, see "Viewing the Results of BI Lineage Scans" on page 252.

Administering the BI Lineage Plug-in

Running and Scheduling Scans

BI Lineage scans can be run only by an unrestricted administrative user (that is, a user who is in the **SAS Metadata Server: Unrestricted** role).

To run a scan, go the **Plug-ins** tab in SAS Management Console, right-click the **BI Lineage** node, and select **New Scan**. In the New Scan Wizard, specify a name for the scan. Then navigate to a starting location, which can be any folder (other than the root folder) in the SAS Folders tree. The scan examines the reports and information maps that are stored in that folder and all of its subfolders, and identifies connected objects regardless of their location.

Because the lineage information is not generated in real time, it is important to keep the scan information updated. To make this task easier, you can create jobs and then schedule them to run at regular intervals. The plug-in can generate jobs for running, exporting, or deleting BI Lineage scans. After the jobs are generated, you can use the Schedule Manager plug-in to schedule the jobs. For details about these tasks, see the BI Lineage plug-in Help in SAS Management Console.

Granting Users Permission to View Scan Results

Scan results are stored in a special metadata repository called the BILineage repository. This repository is created automatically the first time an unrestricted administrative user logs on to SAS Management Console. The BILineage repository should not be used for any purpose other than storing scan results.

To give users permission to view scan results, you must update the BILineage repository's Default ACT to grant ReadMetadata permissions. Follow these steps to grant permission:

1. Log on to SAS Management Console as an unrestricted administrative user. On **Plug-ins** tab, select **BILineage** from the Repository drop-down list.

2. Navigate to **Environment Management** ⇨ **Access Control Templates** ⇨ **Default ACT**, and open the properties window for the Default ACT.

3. Select the **Permission Pattern** tab, and update the information as follows.

   • If you want to give all registered users access to scan results, then grant ReadMetadata permission to the SASUSERS group.

   • If you want to give access to a specific group of users, click **Add** and select the group. When you add the group, the Authorization Manager automatically grants it ReadMetadata permission.

Note: You cannot provide access by setting permissions on the BILineage folder that appears in the SAS Folders tree, because scan results are not stored in the folder.

Viewing the Results of BI Lineage Scans

To view scan results, expand the node for the BI Lineage plug-in, and then expand the node for the scan whose results you want to view.

Within the scan, expand the node for a particular object type (for example, Reports). Then select the folder that contains the object whose lineage information you want to view, as in the following example:
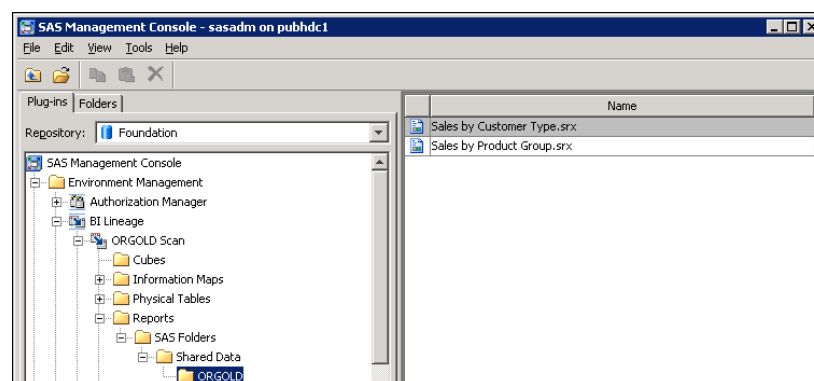
In the right pane, right-click the object whose lineage information you want to view, and select one of the following options:

Lineage
identifies the objects that contribute to a particular report or information map (including objects that are located outside of the scanned folder and its subfolders). For example:

- The lineage information for a report identifies all of the information maps, tables, cubes, columns, filters, and stored processes that contribute to the report.

- The lineage information for an information map identifies all of the tables, cubes, columns, filters, and stored processes that contribute to the information map.

Reverse Lineage
identifies the objects within the scope of the scan that might be affected if a selected object were to change. (Affected objects that are stored outside of the scanned folder and its subfolders are not identified.)

For example, the reverse lineage information for a table shows information maps and reports that use the table (excluding those that are stored outside of the scanned folder and its subfolders). You can use this information to determine which information maps and reports might be affected if the table were to change.

Properties
displays information about the object, as well as a list of the component objects that are members of the selected object (for example, the columns in a table, the data items in a cube, or the sections in a report).

Right-click any of the component objects to display a menu of options for that object. The options include the following:

High-level Lineage
identifies the high-level objects that contribute to a particular component object (such as a data item, column, or report section).

For example, high-level lineage for a report section shows all information maps and tables that contribute to the report section.

High-level Reverse Lineage
identifies the high-level objects within the scope of the scan that would be affected if the selected component object were to change. (Affected high-level objects that are stored outside of the scanned folder and its subfolders are not identified.)

For example, the high-level reverse lineage information for a data item shows all reports that use the data item (excluding reports that are stored outside of the scanned folder and its subfolders). You can use this information to determine which reports would be affected if the data item were to be changed.

Low-level Lineage
identifies other low-level member objects (such as data items, columns, and report sections) that contribute to a particular component object.

For example, low-level lineage information for a data item would show the column or columns that contribute to that data item.

Low-level Reverse Lineage
identifies other low-level objects within the scope of the scan that would be affected if a particular component object (such as a data item, column, or report section) were to change. (Affected objects that are stored outside of the scanned folder and its subfolders are not identified.)

For example, low-level reverse lineage for a data item shows report sections that would be affected if that data item were to change (excluding report sections that are stored outside of the scanned folder and its subfolders).

Properties
displays the general properties for the object.

You should be aware of the following limitations when using scans produced by the BI Lineage plug-in:

- Reverse lineage and low-level reverse lineage information only identifies objects that are stored within the scanned folder and its subfolders. Objects that are stored outside the scope of the scan are not identified. For example, reverse lineage for a table shows reports and information maps that might be affected if the table were to change, excluding reports and information maps that are stored outside of the scanned folder and its subfolders.

  (By contrast, lineage and low level-lineage information includes all objects, regardless of their location in the SAS Folders tree. For example, the lineage information for a report includes all tables and information maps that contribute to the report, including those stored outside the scope of the scan.)

- If the data source for an information map is a table that was created by the SAS Information Maps LIBNAME engine, then only partial lineage information is provided for the map. The lineage information includes the table that the engine created. However, it does not include the original map that the engine accessed to create the table, nor does it include the table that was used to create the original map. Here is an example scenario:

  - Table A was used as a data source for Map B.

  - Map B was used as a data source by the Information Maps LIBNAME engine to create Table C.

  - Table C was used as a data source to create Map D.

  For Map D, the BI Lineage scan would provide the following lineage information: **Map D** ⇨ **Table C**. The lineage information would not include Map B or Table A.

  Note: The SAS Information Maps LIBNAME engine provides Read-Only access to data in an information map. For more information, see *Base SAS Guide to Information Maps*.

- Prompts that are used by information maps are not included in scans.

- Low-level lineages are not supported for sections of reports that directly use tables or cubes. Low-level lineages are supported only for report sections that are based on information maps.

- BI Dashboards and their component parts (such as ranges and indicators) are not included in scans.

For more details about using the BI Lineage plug-in, see the plug-in Help in SAS Management Console.

Suggested Uses for the BI Lineage Plug-in

The BI Lineage plug-in is useful in the following types of situations:

- A report consumer who wants to know where the data on a report originated can use lineage information to identify the information maps, cubes, and tables that provided the source data, as well as any stored processes that processed the data.

- If you modify a table, cube, or information map, you can use reverse lineage information to identify associated objects that might need to be retested or verified. For example, you can identify which information maps and reports (within the scope of the scan) use a particular table. In addition, you can use low-level reverse lineage information to identify which information maps, reports, or report sections (within the scope of the scan) use a particular column.

- For project documentation, you can use the BI Lineage plug-in to create a printable list of the objects in the project and their relationship to one other.

- BI Lineage can provide usage information to help set priorities and make support decisions. For example, you can find out how many cubes, information maps, or reports (within the scope of a scan) use a particular set of tables or a particular data column.

  Note: The BI Lineage plug-in does not report the number of times that a particular object was accessed. This type of usage information can be obtained through tools such as the SAS Audit and Performance Measurement package (see **http://support.sas.com/rnd/emi**) and the Key User Action Log in SAS Web Report Studio (see "Manage the Key User Action Log File" in *SAS Intelligence Platform: Web Application Administration Guide*).

- When troubleshooting security issues, you can use lineage information to identify which objects' permissions to examine. For example, if a user is having difficulty accessing a report, you can use lineage information to identify the information maps, tables, and cubes that the report depends on. To have full access to the report, the user must have the appropriate permission to the metadata and the physical data at each of these levels.

See Also

Chapter 26, "Using the Batch Relationship Reporting Tools," on page 347

Customizing Your Folder Structure

Guidelines for Creating Custom Folders

When you install SAS, a default folder structure is provided. This structure segregates system information from business information, provides personal folders for individual users, and provides an area for shared data. Within this overall structure, you can create a customized folder hierarchy that meets the information management, data sharing, and security requirements of your organization.

You can set up your folder structure under the **Shared Data** folder that is provided, or you can create additional folders under **SAS Folders** in which to store shared content.

Here are some guidelines for setting up a folder structure:

Keep the folder structure as simple as possible.
  A simple folder structure makes your system easier to navigate and easier to administer. You should create new folders or levels of folders only if they provide a specific benefit for your organization.

Develop a folder structure that reflects the organization of your work.
  Folders should be organized the way your work is organized. For example, your folder structure could reflect the following:

  - your company's internal organization. For example, each division or department could have its own high-level folder.

  - types of business activities. For example, you could have a separate folders for human resources, sales, research and development, and marketing.

- geography. For example, each country, sales region, or regional office could have its own folder.

- categories of products. For example, each product line or product group could have its own folder.

- time periods. For example, you could have a folder for each year, quarter, or month.

- categories of users. Generally, this type of folder structure is necessary only in large organizations that have a clear separation of responsibilities (for example, separate teams for data preparation, map creation, and report creation).

- change-control status. If you have just one deployment of the SAS Intelligence Platform (instead of separate deployments for development, test, and production), then you might want to use folders to separate production-status content from content that is in the development or testing stage. To do so, you can set up separate sets of folders for development, test, and production. Then, you can use the promotion tools to move content from development to test and from test to production. See "Example 3: Separate Folder Structures for Development, Test, and Production" on page 258.

Do not set up folders based on SAS client applications.
It is not necessary or desirable to organize objects based on which SAS client applications were used to create them. Organizing folders on this basis can complicate administration tasks such as the assignment of permissions.

Do not set up folders based on object types unless it is necessary for access control.
Organizing folders based on object types can complicate administration tasks such as the assignment of permissions. As a general rule, you should avoid setting up folders on this basis.

Different types of objects such as libraries, information maps, reports, OLAP cubes, and stored processes can be stored together in the same folders without causing any problems. Because client applications such as SAS Information Map Studio and SAS Web Report Studio display only the object types that are used in that application, users will see only the object types that they need to see.

An additional advantage of mixing object types is that associations are easier to maintain. For example, if a folder contains objects that are associated with one another, you can rename the folder without breaking the associations.

However, in some large organizations, you might need to use folders to restrict access to certain types of objects. For details, see the next guideline.

Develop a folder organization that reflects the access rules that you want to enforce.
Folders enable you to easily restrict access to content. For example:

- If you want to prevent departments from accessing each other's content, then you can create a high-level folder for each department and apply different permissions to each of the folders.

- If you want to restrict access to sensitive content (for example, content related to a sensitive product line or a business activity such as human resources), then you can create a separate folder for that content and apply a restrictive access control template (ACT).

- If your organization requires a clear separation of content among different categories of SAS users, then you can create separate folders for each group. Generally, this type of folder structure is necessary only in large organizations that have separate teams of SAS users with different job responsibilities.

For example, suppose you have one group of users that works on data preparation tasks (such as creating libraries, tables, and cubes) and another group creates information maps, stored processes, and reports). To ensure that the groups do not interfere with one another's work, you can create a separate folder for each group and apply different permissions to each of the folders.

See "Baseline ACTs" in *SAS Intelligence Platform: Security Administration Guide* for detailed information about creating Access Control Templates (ACTs) for folders and applying the appropriate permissions at each level.

If you have separate environments for development, test, and production, then use the same folder structure across environments.

Using a uniform folder structure will make it easier to promote objects from one environment to another.

Examples of Custom Folder Structures

Example 1: Multiple Lines of Business across Multiple Regions

Here is an example folder structure for an organization that has multiple lines of business across multiple regions. Each region has its own set of data definitions, information maps, stored processes, and reports. Each region and line of business has a single user or team of users that performs all data integration and business intelligence tasks.
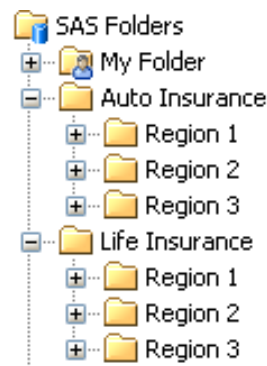
In this scenario, each low-level folder would have permissions that enable WriteMemberMetadata (WMM) access for the SAS users in that specific line of business and region.

Example 2: Multiple Lines of Business with Separate Data Integration and Business Intelligence Users

Here is an example of a folder structure for an organization that has multiple divisions. Within each division, a specific group of SAS users performs data preparation tasks (including the creation of tables and cubes) and another group of users creates information maps, reports, and stored processes. The table and cube definitions are division-wide and are organized by time period. Separate information maps, reports, and stored processes are created for each department.
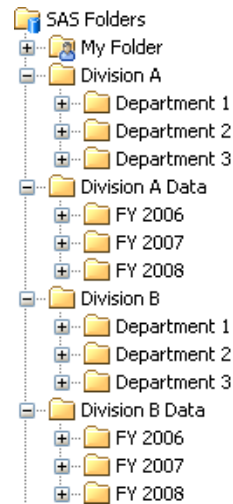
In this scenario, the following permissions would be granted for a division's data folder (for example, the **Division A Data** folder):

- only Read (R) and ReadMetadata (RM) access for users who create maps, reports, and stored processes. These would include users of SAS Enterprise Guide, SAS Information Map Studio, and SAS Web Report Studio.

- WriteMemberMetadata (WMM) access for the users who perform data integration tasks for the division (including users of SAS Data Integration Studio and SAS OLAP Cube Studio).

Example 3: Separate Folder Structures for Development, Test, and Production

As an alternative to creating separate deployments of the SAS Intelligence Platform for development, test, and production, you can use SAS folders to create a change control environment within a single metadata server. The following example demonstrates how to set up folder structures for test and development that mirror your production folders:
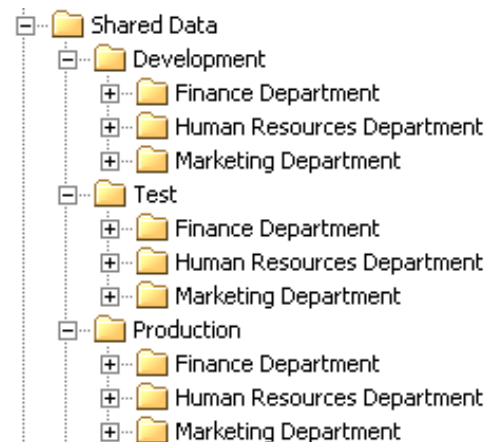
In this scenario, you would give development and testing personnel the appropriate access to the **Development** and **Test** subfolders, and you would give business users the appropriate access to the **Production** subfolders.

You can then use the Export SAS Package and Import SAS Package wizards to move metadata and associated content from **Development** to **Test** and from **Test** to

`Production`. For details, see . To facilitate promotion, use the same folder structure within each of the three folders.

Chapter 19

Managing Setinit (License) Information in Metadata

Updating Setinit Information in Metadata

The metadata repository stores setinit (license) information for some SAS solutions that depend on the SAS middle tier. For a list of these solutions, see http://support.sas.com/kb/49/750.html.

When you receive a new SAS installation data file (SID file) for these solutions, you must use SAS Deployment Manager to update the metadata with new the setinit information. For instructions, see:

- *Software License Renewal Instructions—SAS for Microsoft Windows for x64 Planned Deployment*

- *Software License Renewal Instructions—SAS for UNIX Environments Planned Deployment*

These documents are available at http://support.sas.com/documentation/installcenter/94/index.html.

Viewing Setinit Information In Metadata

You can use SAS Management Console to view setinit information that is stored in metadata. This feature is effective with SAS 9.4M2. Follow these steps:

1. Log on to SAS Management Console as a user with the appropriate credentials. See "Who Can Do What in the Metadata Manager Plug-in" on page 40.

2. On the **Plug-ins** tab, expand the **Metadata Manager** node.

3. Right-click the **Metadata Utilities** node, and select **View metadata setinit details**.

   A dialog box appears, listing the expiration dates for each product. If the repository contains setinit information for more than one site, each site's information is displayed in a separate tab.

4. Click **Save to File** if you want to save the displayed information to a text file.

Part 7

Using the Promotion Tools

Chapter 20
Promotion Tools Overview

Introduction to the Promotion Tools

About Promotion

Promotion is the process of copying selected metadata and associated physical content within or between planned deployments of SAS software. This process is repeatable for a particular deployment.

About the Promotion Tools

The SAS Intelligence Platform provides tools that enable you to promote individual metadata objects or groups of objects from one SAS deployment to another, or from one folder location to another within the same deployment. You can promote metadata as well as the physical files that are associated with metadata.

The promotion tools include the following:

• the Export SAS Package and Import SAS Package wizards, which are available in the **Folders** tab of SAS Management Console. The same wizards are also available in SAS Data Integration Studio and SAS OLAP Cube Studio.

• the batch import tool and the batch export tool, which enable you to perform promotions on a scheduled or repeatable basis. These tools provide most of the same capabilities as the Export SAS Package and Import SAS Package wizards.

See Also

A different promotion tool is provided for portal content. See "Promoting Portal Content" in *SAS Intelligence Platform: Web Application Administration Guide*.

When to Use the Promotion Tools

The promotion tools (including the Export SAS Package and Import SAS Package wizards and the batch export and import tools) support the following types of scenarios:

• promotion of content from one location in the SAS Folders tree to another location in the same tree. For example, you might want to promote newly developed (or newly modified) content from a user's home folder to a location under the **Shared Data** folder.

• promotion of content from a test environment to a production environment. For example, you might want to promote newly developed (or newly modified) content from a metadata server that is installed on a test machine to a metadata server that is installed on a production machine.

• importing content that was exported from a SAS 9.2 or SAS 9.3 environment. This capability is useful in the following situations:

  • You are upgrading from SAS 9.2 or SAS 9.3 to SAS 9.4, but you chose not to use the SAS Migration Utility. For details, see *SAS Intelligence Platform: Migration Guide*.

  • You are running SAS 9.4 in parallel with SAS 9.2 or SAS 9.3, and you want to update your SAS 9.4 system with new content that you created in the earlier release.

  Note: If you need to promote content to support an upgrade from SAS 9.1.3 to SAS 9.4, then please contact SAS Technical Support for more information, and visit the SAS Migration Utility Focus Area at http://support.sas.com/rnd/migration/utility/index.html.

• creating a backup of specific folders and objects. In this case, you would use the Export SAS Package wizard or the batch export tool to create a package file. If you later need to restore the folders or objects, you would use the Import SAS Package wizard or the batch import tool to restore the contents of the package file to the metadata server.

Types of Objects That Can Be Promoted

You can use the promotion tools to promote almost all types of content that are stored in the SAS Folders tree. The content types include libraries, tables, jobs, cubes, information maps, reports, and many other types depending on which applications you have installed. You can also promote security objects (users, user groups, roles, and access control templates), as well as all of the server types that can be registered in metadata. Virtual folders called **Security** and **Servers** are displayed under the **System** folder in the SAS Folders tree in SAS Management Console for use in promoting these objects. In addition, you can promote secured data folders, secured library objects, and secured table objects that have been created to support metadata-bound libraries.

Limitations of the Promotion Tools

The following limitations apply to the promotion tools that are described in this documentation:

*   When using the export and import wizards in SAS Data Integration Studio and SAS OLAP Cube Studio, you can promote only the objects that pertain to the application. For example, in SAS Data Integration Studio, you can promote objects such as libraries, tables, jobs, and cubes, but you cannot promote objects that are unrelated to data integration (such as reports and dashboard objects).

*   You cannot use the export and import wizards or the batch export and import tools to promote portal pages, page templates, portlet instances, and the portal's application and user permissions trees. This content is not stored in the SAS Folders tree. To promote these items, you can use the content promotion tool for the SAS Information Delivery Portal. See "About the Portal Promotion Tools" in *SAS Intelligence Platform: Web Application Administration Guide*.

Overview of the Promotion Process

Promotion is a three-step process:

*   First, you must prepare the source and target environments. The preparation steps include ensuring that the required servers are configured and running. If you are promoting metadata from one metadata server to another, you must prepare the target environment by performing tasks such as creating directories for the physical files. For details, see "Preparing to Use the Promotion Tools" on page 271.

*   Second, you use the Export SAS Package wizard (or the batch export tool) to export the selected objects or folders from the source metadata server. The wizard or the batch tool places the exported objects in a SAS package (SPK) file. If you want the SPK file to be available remotely, then you can direct the package to a shared location within your file system. You can also transport the package by email or FTP after it has been created.

*   Third, you use the Import SAS Package wizard (or the batch import tool) to import the package file. You can import the package to either of the following:

    *   another metadata server, which can be located either on the same host machine or a different host machine

    *   a different location in the SAS Folders tree on the source metadata server (except for server and security metadata, which can be imported only to the **Server** and **Security** folders, respectively)

Figure 20.1 Three-step Promotion Process
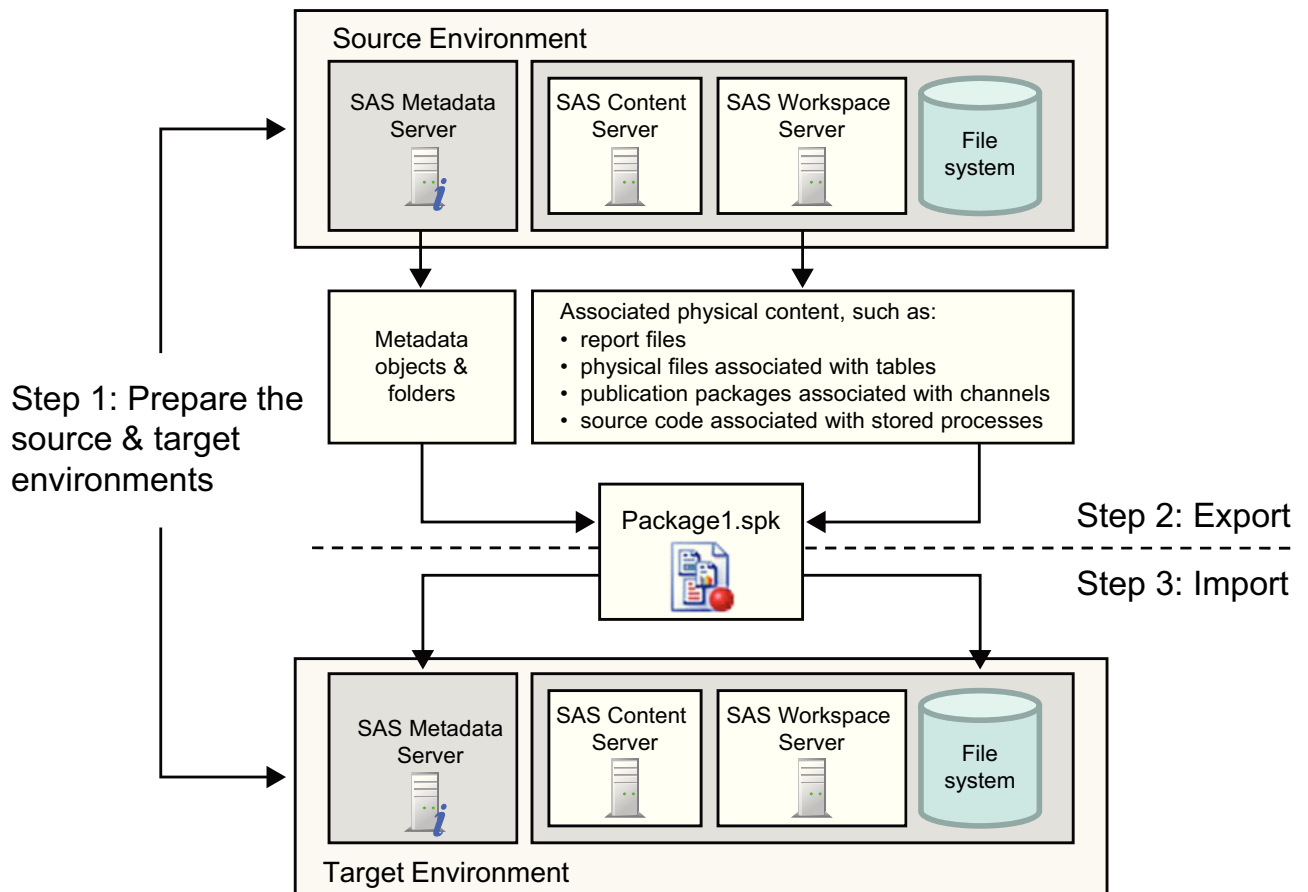
The package format is the same regardless of whether the package is created in SAS Management Console, in SAS Data Integration Studio, in SAS OLAP Cube Studio, or by the batch export tool. In addition, the package format is the same regardless of the host machine's operating system. Therefore, packages can be interchanged as follows:

*   A package that is exported by using SAS Management Console can be imported by using SAS Data Integration Studio or SAS OLAP Cube Studio, and vice versa. However, SAS Data Integration Studio and SAS OLAP Cube Studio can export and import only the objects that pertain to the application.

*   A package that is exported by using the Export SAS Package wizard can be imported by using the batch import tool.

*   A package that is exported by using the batch export tool can be imported by using the Import SAS Package wizard.

*   You can export a package from a Windows system and import it to a UNIX system, and vice versa.

Features of the Promotion Tools

Selecting Objects to Export

The Export SAS Package wizard and the batch export tool provide the following options for specifying which objects are to be exported:

- You can select multiple nested folders.

- You can select **Include empty folders** if you want to export folders that do not contain any subfolders or objects. (Otherwise, the empty folders are not exported.)

- You can include all of the objects that are contained in a folder or just selected objects.

- You can filter content, and you can identify and include dependent objects, as described in the next two topics.

Filtering Content

The export wizard includes a Filter feature that enables you to select objects based on the object name, object description, object type, or the time period during which the object was created or last modified. The Advanced Filter Options enable you to identify objects for export based on their keywords, notes, extended attributes, and responsibilities fields. The same filtering capabilities are included in the batch export tool.

Identifying and Including Dependent Objects

> *CAUTION:*
> **In order for objects to function properly in the target environment, you must import the resources that objects depend on, unless those resources already exist in the target environment.** For example, if you want reports to function properly, the information maps that the reports depend on must be present. If a report has stored processes or images associated with it, then those objects must be present in the target system.

The BI Lineage tool can assist you in identifying associations between objects. In addition, the promotion tools provide the following features to help you ensure that you include all of the necessary objects:

- You can use the **Include dependent objects when retrieving initial collection of objects** option of the export wizard (or the **-includeDep** option of the export batch tool) to specify that all of an object's dependencies be identified recursively and included in the export. For example, if you are promoting a report, this feature identifies and automatically exports objects such as information maps, cubes, tables, and libraries that the report depends on.

- In the export wizard, the **Dependencies** tab lists the highlighted object's direct dependencies, and the **Used by** tab identifies objects that directly use the highlighted object. You can then specify which of the identified objects to include in the export. After these objects are added to the list of objects to export, you can view the new objects' **Dependencies** and **Used by** tabs to identify additional associated objects to export.

For example, suppose the **Dependencies** tab for a report lists an information map. You can select the map to add it to the list of objects to export. You can then highlight the map and display its **Dependencies** tab, which lists any libraries, tables, and cubes that the map depends on. These objects can then be selected to be added to the export.

Note: The **Used by** and **Dependencies** tabs do not display information for all object types.

Promoting Physical Content along with Metadata

For most object types, the export and import wizards (and the batch export and import tools) promote both the physical content and the metadata. For some object types, the wizards enable you to select whether to include or exclude physical content. For example, when you export a stored process, you can choose to include or exclude the associated source code files. When you export a table or external file, you can choose to include or exclude the associated physical file.

When you use the batch tools, the promotion automatically includes all associated content except physical files for tables and external files. However, you can use a global option to prevent the batch import tool from importing source code files for stored processes.

For a detailed list of the types of physical content that can be promoted, see "Types of Physical Content That Are Promoted with Metadata Objects" on page 301.

Restoring Metadata Associations

Depending on the types of objects that you are importing, you might need to associate imported objects with objects or entities on the target system. The import wizard displays prompts that enable you to establish the needed associations. For example:

• Some objects, such as libraries and physical files, must be associated with a physical directory on the target system. Physical directories cannot be promoted. Therefore, the import wizard prompts you to specify an already existing physical directory on the target system.

• Some objects, such as libraries and stored processes, must be associated with a SAS Application Server on the target system. When you import these objects, the import wizard prompts you to select the appropriate application server on the target system. (Although it is possible to promote servers, you will generally want to use the servers that already exist on the target system.)

If you are using the batch tools, then you can specify the associations in the substitution properties file.

Reorganizing Content By Moving Objects to Different Folders

The import wizard and the batch import tool enable you to reorganize content as part of the promotion process. You can do so by specifying new folder locations for individual objects that you are importing. The connections among objects are maintained.

In the batch import tool, you can change the location of an object by specifying a new property for the object in the substitution properties file. The new folder location must already exist in metadata when you use this method.

Resolve Column Differences When Importing Tables and External Files

When you import metadata for a table or external file, and a table or external file of the same name already exists in the target folder, you can use the Change Analysis feature of the import wizard to identify and resolve differences in column metadata.

In the batch import tool, you can use global options to globally control the handling of column differences.

Scheduling Flows on Import

The Schedule Manager of SAS Management Console contains a 'Schedule the Flows' option that you can use when importing a package file that contains a flow.

This option is not checked by default. When it is checked, package files that contain a flow will automatically have their jobs rescheduled when imported.

Here are key points about using this option:

*   If the scheduler is not available or the connection cannot be made, then the flow is not scheduled.

*   Scheduling flows during import is not available when importing in batch mode.

Preparing to Use the Promotion Tools

Ensure That the Required Servers Are Present

Before using the export and import wizards (or the batch export and import tool), make sure that the appropriate servers exist and are running, as follows:

*   The object spawner must be running in both the source environment and the target environment.

*   If you are promoting physical content that is associated with stored processes, tables, jobs, libraries, or external files, then a standard (non-pooled) workspace server must exist in both the source environment and the target environment.

*   If your stored processes create packages that are stored permanently on the SAS Content Server, then a SAS Content Server must exist and must be running in both the source environment and the target environment.

*   If you are promoting reports and want to store report content on a SAS Content Server, then a SAS Content Server must exist and must be running in both the source environment and the target environment.

*   If you are promoting visual data queries, make sure that the appropriate instance of the SAS Web Application Server is running. For example, SAS Visual Analytics might be using an instance called SASServer12.

Ensure Appropriate User Credentials

Make sure that the user who launches the wizard (or whose credentials are specified in the batch export or import command) has the following credentials and permissions:

- ReadMetadata access to the objects that are being exported

- WriteMetadata access to the objects that are being imported

- WriteMemberMetadata access to the folders where objects are imported

- appropriate access to associated physical content (for example, tables and source code files) that is being exported or imported

- credentials to access the workspace server, if you are exporting or importing physical content that is associated with stored processes, tables, jobs, libraries, or external files (see "Requirements for Accessing a Standard Workspace Server" on page 43)

  Note: If your user account does not have workspace server credentials, the wizard prompts you to enter additional credentials. For example, you will be prompted to enter additional credentials if you are running the wizard using an internal account (a user ID with the suffix @saspw).

Ensure Adequate Memory

If you are planning to export or import a large number of sizable objects, then you might need to increase the amount of memory that is allocated to the import or export process. Information maps, SAS Data Integration Studio jobs, and tables are the object types that most frequently require additional memory. To increase allocated memory, follow these steps:

1. On the client machine where you are executing the export or import, open the appropriate INI file in a text editor as follows:

   - If you are using SAS Management Console, open sasmc.ini, which is located in *SAS-installation-directory***SASManagementConsole\9.4**.

   - If you are using SAS Data Integration Studio, open distudio.ini, which is located in *SAS-installation-directory***SASDataIntegrationStudio\4.7**.

   - If you are using SAS OLAP Cube Studio, open olapa.ini, which is located in *SAS-installation-directory***SASOLAPCubeStudio\4.4**

   - If you are using the batch import or export tool, then open ImportPackage.ini or ExportPackage.ini. These files are located in *SAS-installation-directory***SASPlatformObjectFramework\9.4**.

2. Locate the following line:

   JavaArgs_1=-Xmx512m

3. Replace **512** with a higher value (for example, **1024**), and save your changes.

   Note:

   - For SAS Management Console and SAS Data Integration Studio, this value is set to **1024** by default. For the batch tools and SAS OLAP Cube Studio, the default value is **512**.

   - If your desktop applications run in the Citrix application delivery environment, then you can increase the size of your Java heap to the Citrix

limit. All versions of Citrix support a maximum Java heap size of 512 megabytes, and some versions support higher limits.

As an alternative to increasing memory, you can perform multiple imports or exports that include a smaller number of objects instead of importing or exporting all of your objects at the same time.

Set Up Security for Folders and Objects

In some cases, you might want to set up and test access controls for folders or objects in the source environment, and then include these access controls in the promotion. To use this approach, see "Promoting Access Controls" on page 292.

As an alternative, you might want to set up folders on the target metadata server and apply access controls to them before you import your metadata.

Create Directories for Physical Files on the Target System

On the target system, set up directories for the physical content that you plan to promote with the metadata. For example, you might need to create directories for the following:

- physical locations of libraries

- physical files and external format files (EFFs) that are associated with external files

- custom code that is associated with jobs, if the code is stored outside of the metadata

- source code repositories for stored processes

- **File or directory** prompts for which a default file or directory value is specified (see "Promoting Prompts" on page 295)

The user who will execute the import should have Write access to these directories.

About the Promotion Tools Documentation

To learn to use the promotion tools, access the following additional documentation:

- Chapter 21, "Using the Export SAS Package and Import SAS Package Wizards," on page 275

- Chapter 22, "Reference Information for Promotion Tools," on page 291, which provides important supplemental information that applies to the wizards as well as the batch export and import tools. The following information is included:

  - details about promoting specific object types

  - the types of physical content that are promoted along with metadata

  - considerations for promoting metadata and physical content from previous versions of SAS

  - troubleshooting information

- Chapter 25, "Using the Batch Export and Import Tools," on page 325 in the SAS Intelligence Platform Batch Tools section of this guide.

- the product Help for SAS Management Console, SAS Data Integration Studio, or SAS OLAP Cube Studio

Chapter 21

Using the Export SAS Package and Import SAS Package Wizards

About the Export SAS Package and Import SAS Package Wizards

The Export SAS Package and Import SAS Package wizards provide a graphical user interface for performing promotions. The wizards are available in SAS Management Console, SAS Data Integration Studio, and SAS OLAP Cube Studio. You can use the wizards to promote individual metadata objects or groups of objects from one metadata server to another, or from one location to another on the same metadata server. You can also promote the physical files that are associated with the metadata.

This chapter tells how to access the wizards. It then provides an example scenario that shows how to use the wizards to export and import a package containing several types of related business intelligence objects. The chapter also provides instructions for using the advanced features of the wizards.

As you use the wizards, be sure to refer to the Chapter 20, "Promotion Tools Overview," on page 265 and the Chapter 22, "Reference Information for Promotion Tools," on page 291. You can also refer to the product Help for SAS Management Console, SAS Data Integration Studio, or SAS OLAP Cube Studio.

Accessing the Export SAS Package and Import SAS Package Wizards

To access the Export SAS Package Wizard in SAS Management Console, SAS OLAP Cube Studio, or SAS Data Integration Studio:

1. Right-click any folder or object. (In SAS Management Console, you must first open the **Folders** tab.)

2. In SAS Management Console or SAS OLAP Cube Studio, select **Export SAS Package**. In SAS Data Integration Studio, select **Export ⇨ SAS Package**.

To access the Import SAS Package Wizard in SAS Management Console, SAS OLAP Cube Studio, or SAS Data Integration Studio:

1. Right-click the folder under which you want to import the folders and objects.

2. In SAS Management Console or SAS OLAP Cube Studio, select **Import SAS Package**. In SAS Data Integration Studio, select **Import ⇨ SAS Package**.

Example Usage Scenario for Export SAS Package and Import SAS Package Wizards

Description of Scenario

In this example, a set of objects has been developed in Environment 1 and will be promoted to a different metadata server in Environment 2. The objects to be promoted are as follows:

- two reports: Sales by Customer Type and Sales by Product Type

- a stored process called Sales Analysis

- two information maps: Customer Orders and Product Sales

- a library called Orion Star

- three SAS tables: CUSTOMER_DIM, PRODUCT_DIM, and ORDERS_FACT

The metadata for the library, the tables, and the information maps is stored in **/Shared Data/Orion Star Data**, as shown here:
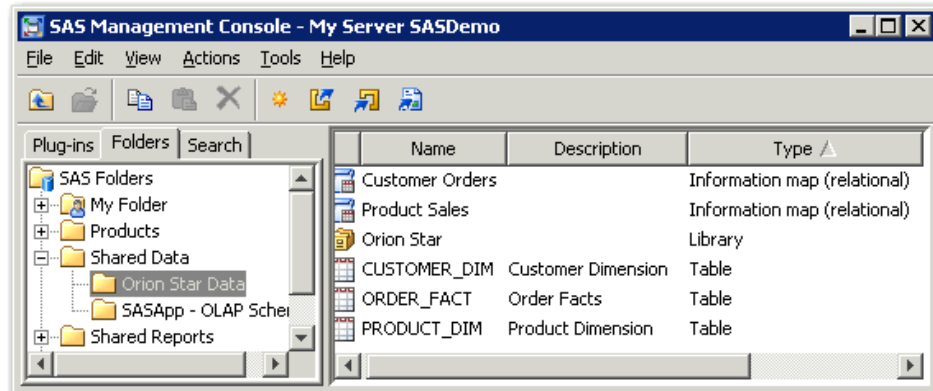
The metadata for the reports and the stored process is stored in **/Shared Reports/ Orion Star Reports**, as shown here:
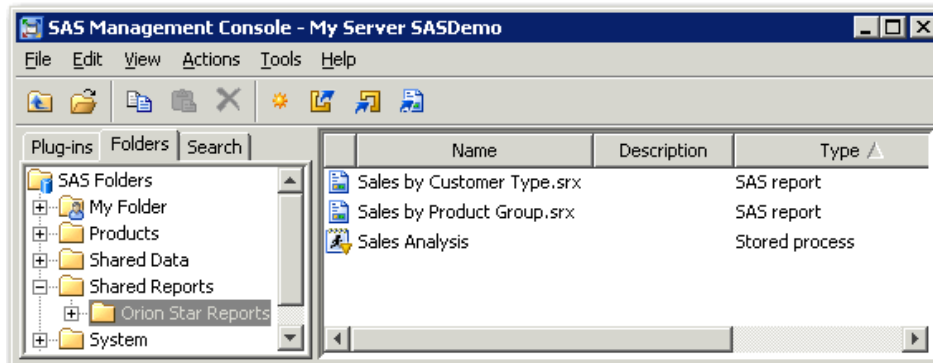
Note: This example scenario uses SAS Management Console to perform the promotion. If you use SAS Data Integration Studio or SAS OLAP Cube Studio, the procedure is the same except where noted.

Before promoting these objects, it is important to understand their associations with one another, as well as their associations with other objects and entities that do not appear in the tree. The following table describes the associations that are involved in the example scenario, as well as the actions that must be taken to ensure that the objects will function properly in the target environment.

Table 21.1   *Example Scenario: Objects to Be Promoted and Their Associations*

Metadata Object or Folder Name	Associated Objects or Content in the Source Environment	Action Needed
Orion Star Reports and **Orion Star Data** folders	Reports_ A ACT (associated with the **/ Shared Reports/ Orion Star Reports** folder) and Data_A ACT (associated with the **/ Shared Reports/ Orion Star Reports** folder)	If you want to include access controls in the import: Promote the ACTs and the user groups to which they refer (or create them in the target environment) before you promote the affected objects. Make sure that the ACTs and groups have the same names in both environments.

Metadata Object or Folder Name	Associated Objects or Content in the Source Environment	Action Needed
Sales by Customer Type report	Sales by Customer Type.srx file (on SAS Content Server)	Before the promotion, ensure that a SAS Content Server exists on the target system.
	Customer Orders information map	Promote the information map and its associated tables with the report.
Sales by Product Group report	Sales by Product Group.srx file (on SAS Content Server)	Ensure that a SAS Content Server exists on the target system.
	Product Sales information map	Promote the information map and its associated tables with the report.
	Sales Analysis stored process	Promote the stored process with the report.
Sales Analysis stored process	Analysis.sas source file	Before the promotion, set up a source code directory in the file system of the target system and apply appropriate permissions.
	Source file repository definition	Before the promotion, ensure that a workspace server exists on the target system.
	Stored process server	Before the promotion, ensure that a stored process server exists on the target system.
Orion Star library	CUSTOMER_DIM, PRODUCT_DIM, and ORDER_FACT tables	Before the promotion, identify a physical location for the library on the target system.
CUSTOMER_DIM, PRODUCT_DIM, and ORDER_FACT tables	Physical tables	Import the physical tables with the table objects.

Step 1: Export the Folders and Objects

For the scenario that is described in the previous topics, you would follow these steps to export the folders and objects from the source environment:

1. Make sure that you have prepared the source environment as described in "Preparing to Use the Promotion Tools" on page 271.

2. Open SAS Management Console, and connect to the source metadata server (that is, the metadata server from which you want to promote the folders and objects).

3. Select the **Folders** tab. (This step is not necessary in SAS Data Integration Studio or SAS OLAP Cube Studio.)

4. Right-click the **SAS Folders** node, and click **Export SAS Package**. (In SAS Data Integration Studio or SAS OLAP Cube Studio, you would right-click the folder that contains the objects that are to be exported.)

5. On the first Export SAS Package page, complete the following steps:

    a. Enter a name for the SPK file that will contain the exported objects (the default name is `Package1.spk`).

        Or you can click **Browse**, navigate to the location on the file system where you want the SPK file to be saved, and specify a name for the file. For convenience, you might want to choose a location that will be accessible from the target environment.

    b. Select **Include dependent objects when retrieving initial collection of objects**, as shown in the following example. This option helps ensure that all objects on which the exported objects depend are included in the export operation.
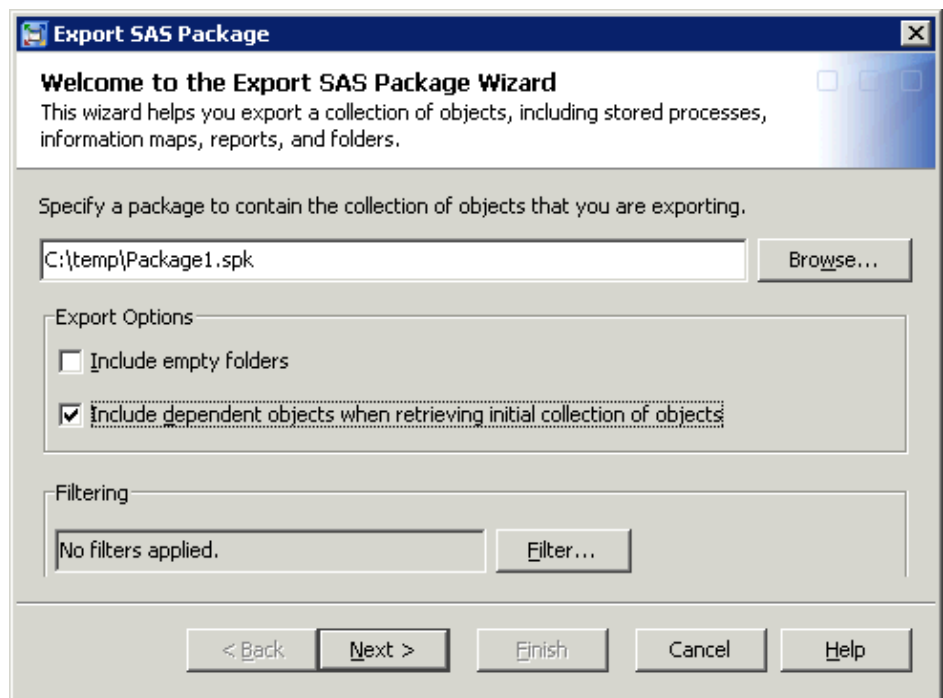
    Note: On this page, you can click **Filter** to select objects based on the object name, the object type, or the time period during which the object was created or last modified. You can also use the **Advanced Filter Options** to select objects based on keywords, notes, extended attributes, and responsibilities fields. (This scenario does not use the filter features.)

    c. Click **Next**.

6. On the next Export SAS Package page, do the following:

    a. Select or clear the appropriate check boxes to specify which folders and objects are to be exported, as shown here. Be sure to select the folders whose access controls you want to promote.
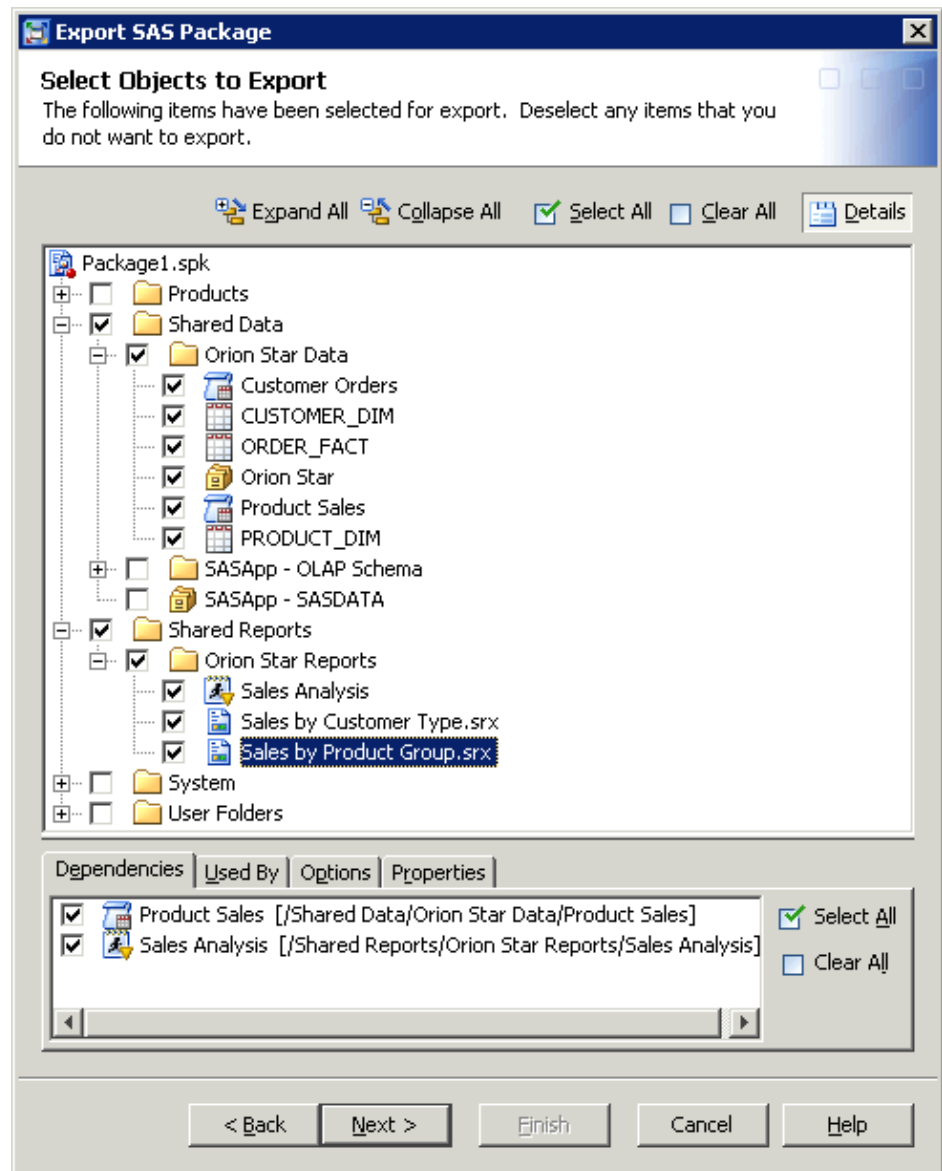
b. Highlight each information map, report, and library object that you have selected. As you highlight each object, view the **Dependencies** tab to ensure that you have included the objects that the highlighted object depends on.

In the preceding example, the report **Sales by Product Group** is highlighted. The **Dependencies** tab lists the **Product Sales** information map and the **Sales Analysis** stored process, which are used by this report. Both objects have been selected to export.

You can also display the **Used By** tab to identify objects that make use of the objects that you have selected.

Note:

• The **Include dependent objects when retrieving initial collection of objects** option (selected on the first wizard page) identifies dependencies only for the objects that are initially selected on this wizard page. If you click **Next** to go to the next wizard page, and you then click **Back** to return to this page and select additional objects, you must use the **Dependencies** tab to identify dependencies for the additional objects.

- If you press the Ctrl key and highlight multiple objects, the
Dependencies tab displays dependencies for all of the highlighted
objects.

- The **Dependencies** and **Used By** tabs do not display information for all
object types.

c. Highlight each of the table objects, and use the **Options** tab to specify whether
the associated physical table is to be exported with the metadata.

In the following example, the `CUSTOMER_DIM` table is highlighted. On the
Options tab, the **Include/replace physical table** check box has been selected to
indicate that the physical table is to be included in the export.
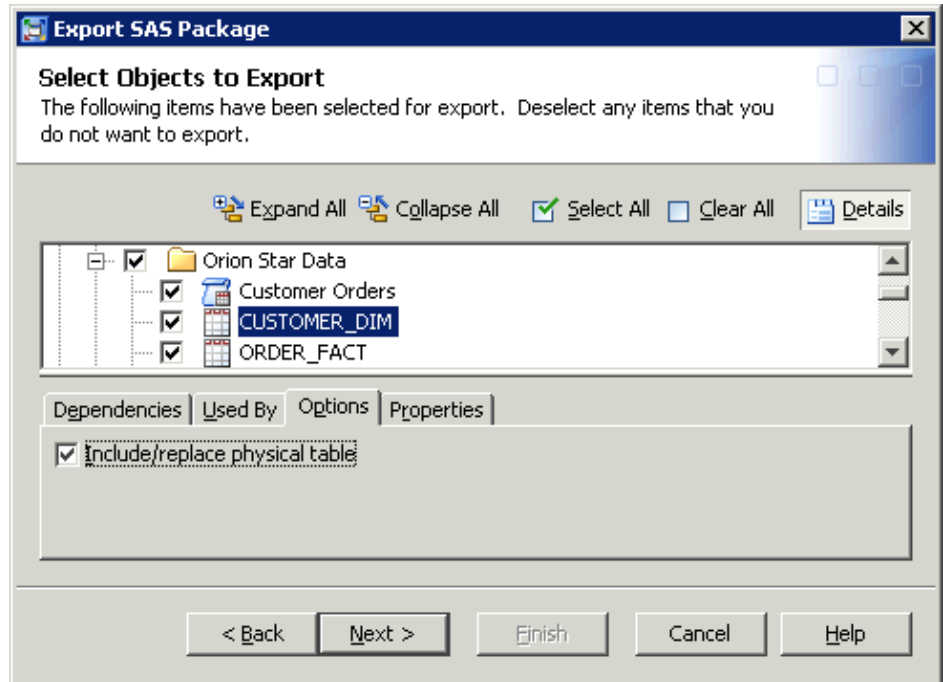
Note:

- If you choose the option to export physical tables, you should be aware
that large data sets could require lengthy processing times (since data sets
are packaged in the SPK file along with the metadata).

- If you press the Ctrl key and highlight multiple objects of the same type
(for example, multiple information maps), the **Options** tab enables you to
select options for all of the highlighted objects.

d. After you have selected all of the appropriate metadata objects, dependencies,
and options, click **Next**.

7. The next Export SAS Package page displays a summary of the objects and folders
that you selected. If the list is correct, click **Next** to begin the export process.

8. As the export process begins, the wizard might display a prompt asking you to log on
to the SAS Application Server. Be sure to log on as a user who has the appropriate
credentials. See "Requirements for Accessing a Standard Workspace Server" on page
43. If have already provided the correct credentials, the prompt does not appear.

9. When the export process finishes, a log with a date-and-time stamp is saved in your
user directory and a message like the following is displayed:
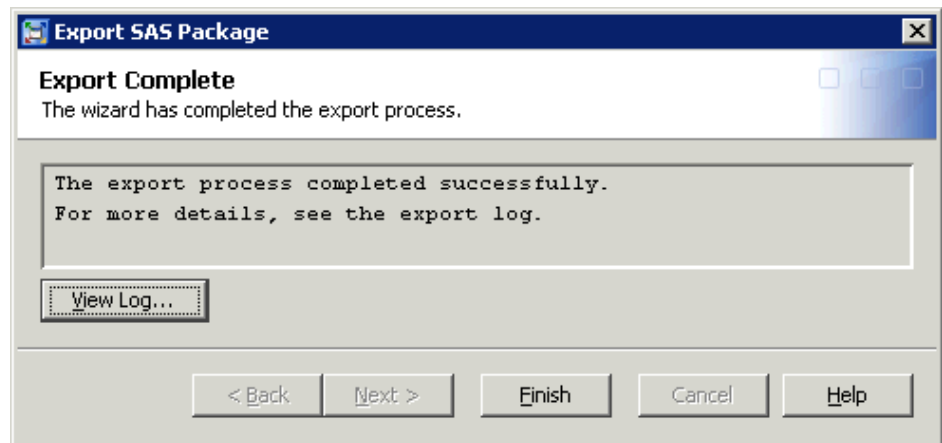
If errors or warnings are present, you can click **View Log** to view explanatory messages. For assistance in resolving errors, see "Special Considerations for Promoting Content to SAS 9.4 from an Earlier SAS Version" on page 303.

Note: For complete step-by-step instructions for using the export wizard, see the product Help for SAS Management Console, SAS Data Integration Studio, or SAS OLAP Cube Studio.

Step 2: Import the SPK File

Follow these steps to import the folders and objects from the SPK file into the target environment. These steps assume that you have completed the export without errors.

1. Make sure that you have completed all of the necessary actions to prepare the target environment. See "Preparing to Use the Promotion Tools" on page 271 and the Action Needed column of Table 21.1 on page 277.

2. Locate the SPK file that was created by the export wizard. If necessary, copy it, move it, or e-mail it to a location that is accessible from the target environment.

3. Open SAS Management Console, and connect to the target metadata server (that is, the metadata server to which you want to promote the folders and objects).

4. Select the folder under which you want to import the folders and objects.

5. Right-click to display the menu, and click **Import SAS Package**.

6. On the first Import SAS Package page, complete the following steps:

   a. Click **Browse**, and navigate to the SPK file that contains the folders and objects.

   b. Select **Include access controls**. This option imports all associated access controls and establishes associations with appropriate users, groups, and ACTs if they exist on the target system.

   c. Select **Preserve source path information for objects in the package**. This option places objects and folders in the same path where they were located in the source environment. If the folders that are included in the path do not exist, they are created. (You can select this option only if you are importing to the **SAS Folders** level.)

   d. Select the **All Objects** radio button. This will cause the imported objects to overwrite any objects of the same name that already exist.

      Note: If you do not want to overwrite objects that already exist, then you would select **New Objects Only**.
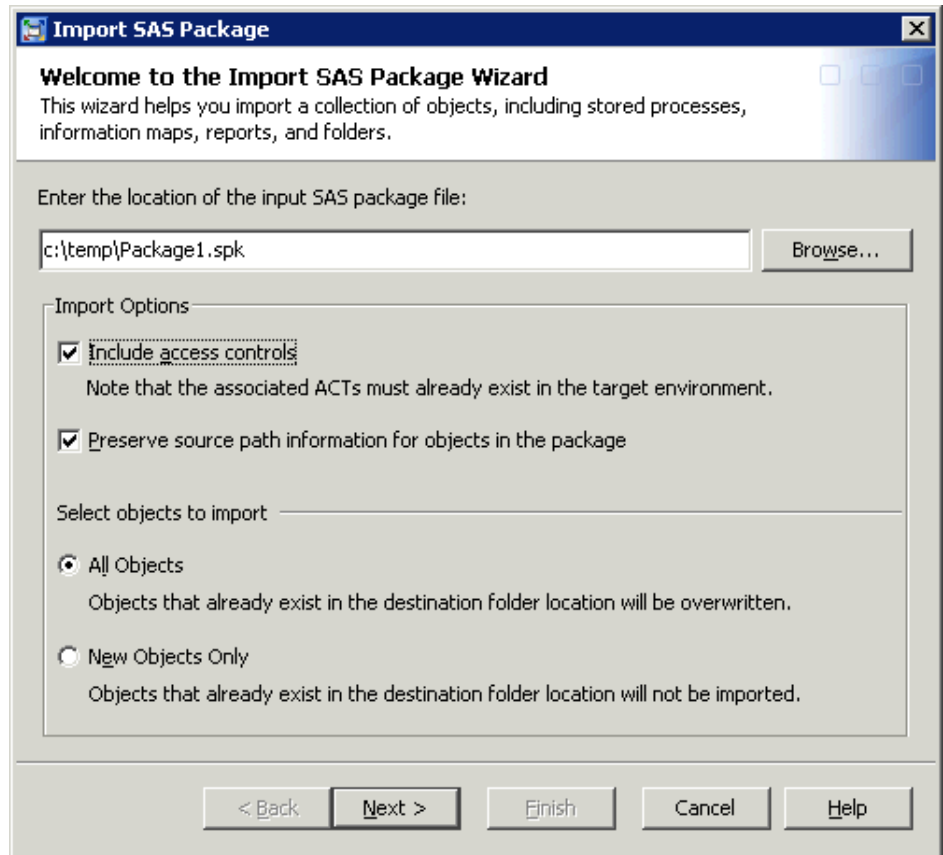
e. Click **Next**.

7. The next Import SAS Package page displays a tree that shows the objects that are included in the package.
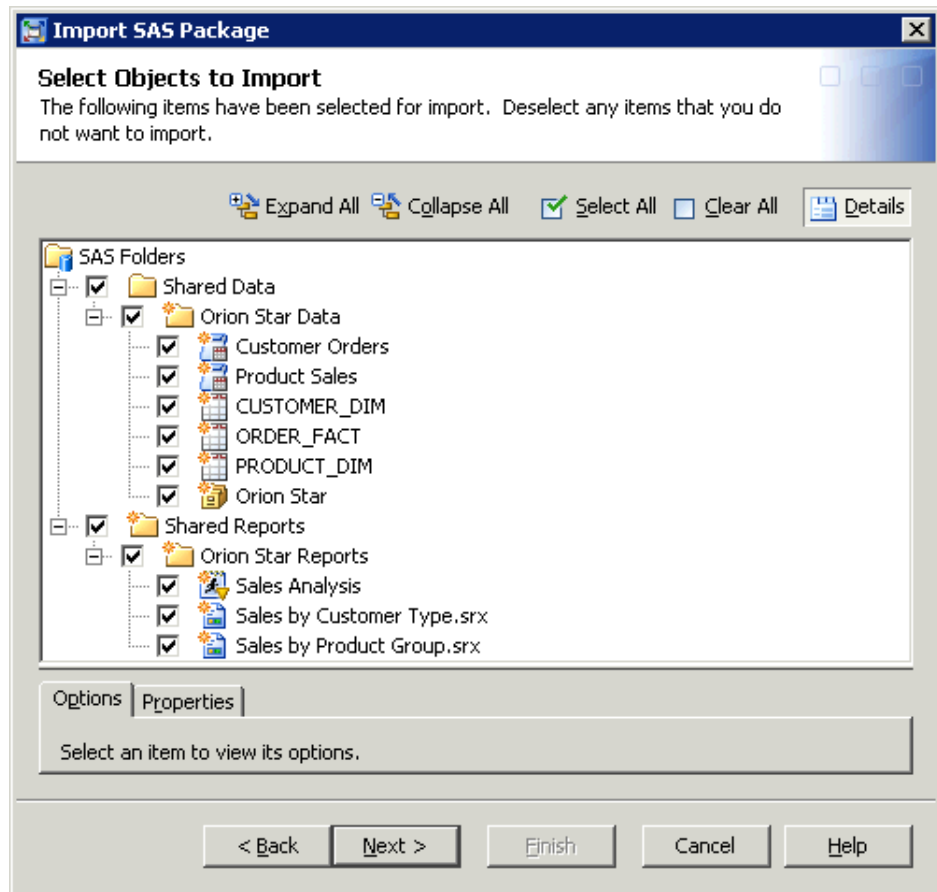
An icon indicates when the user does not have the necessary permissions (⊘) in the target environment, and other icons indicate whether each folder or object is new (✷) or already existing (❗).

On this page, you can highlight one or more table objects, and then use the **Options** tab to indicate whether the physical tables are to be imported along with the metadata. You can also edit the table names. (These options are available only if you exported the physical tables.)

If the tables already exist on the target system, the **Options** tab also enables you to specify how differences in column metadata are to be handled. You can use the Change Analysis feature to resolve individual column differences.

Then click **Next**.

8. The next Import SAS Package page provides advance information about the metadata values that you will need to select on subsequent windows. For this scenario, you will need to select values for an application server, a library directory path, and a source code repository. Click **Next**.

9. In the **Target** column of the next page, select the application server that the imported objects will be associated with in the target environment.
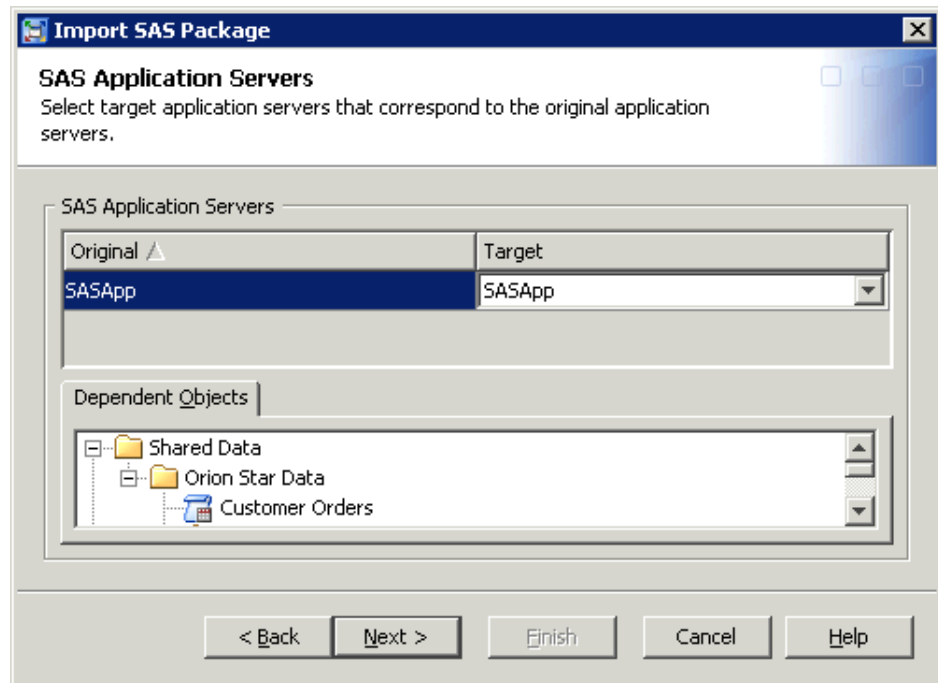
Click **Next**.

10. In the **Target** column of the next page, select the path where the source code file for the stored process should be stored in the target environment. (If the source code repository definition does not yet exist, then click **Add** to create it.) Then click **Next**.

11. In the **Target** column of the next page, enter (or browse to) the path where the library is to be located in the target environment. Then click **Next**.

12. The next page displays a list of the objects that will be imported. The metadata associations for these objects, both in the source environment and the target environment, are also listed. If the information is correct, click **Next**.

13. As the import process begins, the import wizard might display a prompt asking you to log on to the SAS Application Server. Be sure to log on as a user who has the appropriate credentials. See "Requirements for Accessing a Standard Workspace Server" on page 43. If you are already logged on to SAS Management Console, SAS Data Integration Studio, or SAS OLAP Cube Studio with the correct credentials, then the prompt does not appear.

14. When the import process finishes, a log with a date-and-time stamp is saved in your user directory, and a message like the following is displayed:
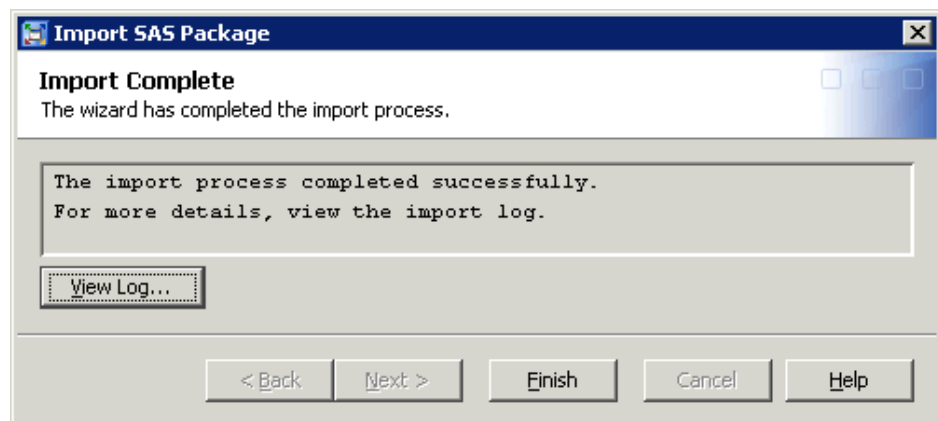
If errors or warnings are present, you can click **View Log** to view explanatory messages. For assistance in resolving errors, see "Troubleshooting the Promotion Tools" on page 305.

*Note:* The preceding steps are for the example scenario. Additional wizard pages might appear, depending on which types of objects you are promoting. For complete step-by-step instructions for using the import wizard, see the product Help for SAS Management Console, SAS Data Integration Studio, or SAS OLAP Cube Studio.

Using Advanced Features of the Import and Export Wizards

Move Objects to Different Folders When Using the Import Wizard

The import wizard and the batch import tool enable you to reorganize content as part of the promotion process. You can do so by specifying new folder locations for individual objects that you are importing. The connections among objects are maintained.

To use this feature, select an object (for example, an information map) on the Select Objects to Import wizard page. Then select the **Properties** tab, as shown here:
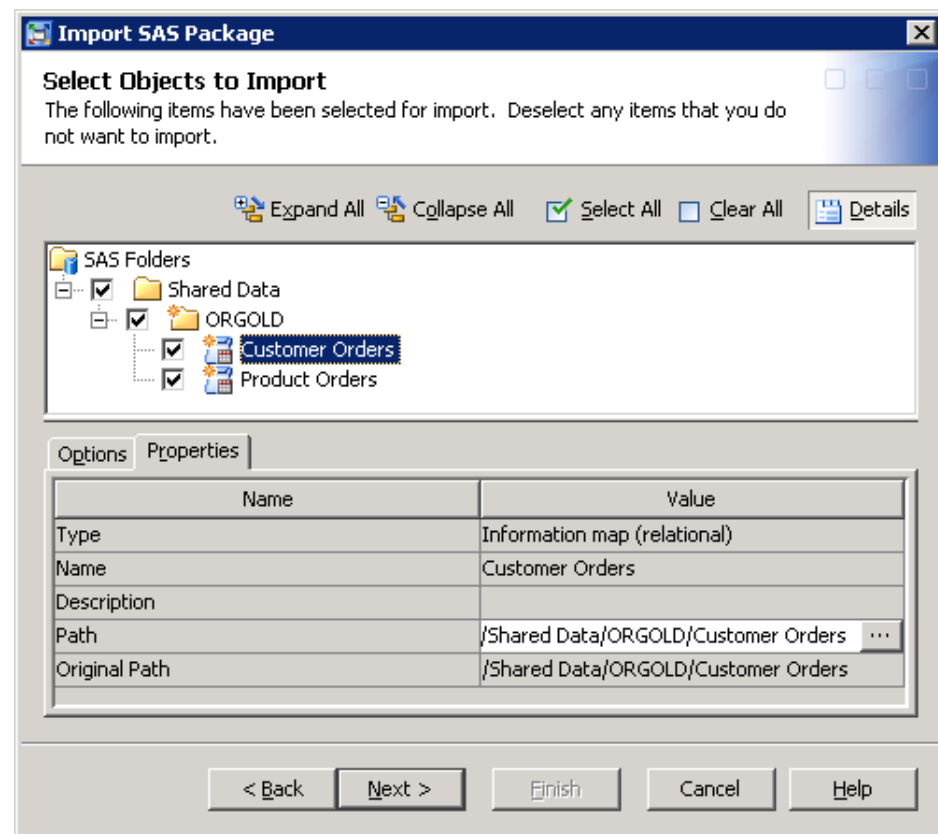
In the Value column of the Path field, click the **Browse** button ( ). When the Select a Location dialog box appears, select a target location in the SAS Folders tree. You can use the **New Folder** button ( ) in this dialog box to create a new folder.

Note:

- You must repeat these steps for each object that you want to move.

- You can use this feature to move folders, including the highest-level folder that you are importing. You cannot move the root folder (**SAS Folders**).

- You cannot move objects from the **Servers** and **Security** virtual folders.

- In the batch import tool, you can change the location of an object by specifying a new property for the object in the substitution properties file. For details, see

Use the Import Wizard's Change Analysis Feature to Resolve Column Differences in Tables and External Files

When you import metadata for a table or external file, and a table or external file of the same name already exists in the target folder, you can use the Change Analysis feature of the import wizard to identify and resolve differences in column metadata. Use the feature to control column handling on either a global or a fine-grained basis, as follows:

Note: This feature applies only if you are overwriting existing objects as a result of selecting the **All Objects** option on the first wizard page.

1. On the Select Objects to Import page, highlight one or more tables or external files that you are importing.

   The wizard compares the target table to the table that is being imported to determine whether any differences exist.

2. If differences are detected, you can specify how column differences are to be handled by default for this import. Select or deselect the following check boxes on the **Options** tab:

   - **Keep target columns not found in the source** is selected by default. This option retains columns in the target table if the columns are not present in the imported table. Deselect this check box if you want to delete columns from the target table if the columns are not also in the imported table. (In the next step, you will be able to override this action for individual columns.)

   - **Include source columns not found in the target** is selected by default. This option adds columns from the imported table if the columns are not present in the target table. Deselect this check box if you do not want to add any new columns to the target table. (In the next step, you will be able to override this action for individual columns.)

3. Click the **Change Analysis** button to view column information for each table that you are importing. In the Change Analysis dialog box, you can update the check boxes as needed to specify how each column is to be handled during the import. Select a column to view its change analysis information, as shown in this example:
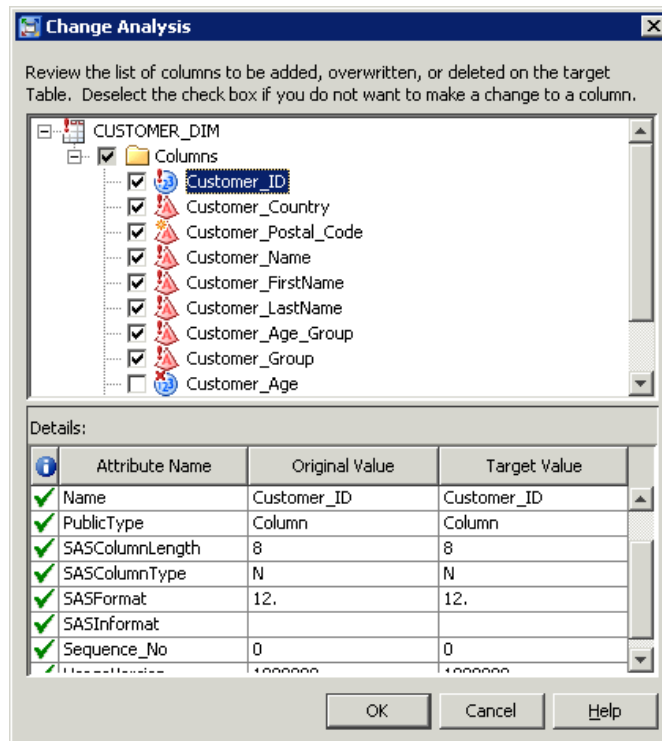
The dialog box might also list items such as table indexes and keys, which cannot be deselected.

The following table explains the meaning of the icons and check boxes in the Change Analysis dialog box:

Icon Explanation	What the Import Will Do If the Check Box Is Selected	What the Import Will Do If the Check Box Is Not Selected
The column metadata exists in the target table but not in the imported table.	The column metadata will be deleted from the target table. If **Keep target columns not found in the source** is deselected on the **Options** tab, the check boxes for these columns are selected by default.	The column metadata will remain in the target table. If **Keep target columns not found in the source** is selected on the **Options** tab, the check boxes for these columns are deselected by default.
The column metadata exists in the imported table but not in the target table.	The column metadata will be added to the target table. If **Include source columns not found in the target** is selected on the **Options** tab, then the check boxes for these columns are selected by default.	The column metadata from the imported table will not be imported. If **Include source columns not found in the target** is deselected on the **Options** tab, then the check boxes for these columns are deselected by default.

Icon Explanation	What the Import Will Do If the Check Box Is Selected	What the Import Will Do If the Check Box Is Not Selected
❗ The column metadata exists in both the imported table and the target table.	☑ ❗ The column metadata from the imported table will overwrite the column metadata for the target table.	☐ ❗ The column metadata will not be imported from the imported table.

For more information about a column, refer to the **Details** section of the dialog box, which displays source and target metadata values for the currently selected column.

See Also

For information about controlling the handling of column differences in the batch import tool, see "Set Global Options for the Batch Export and Import Tools" on page 326.

Chapter 22

Reference Information for Promotion Tools

About the Reference Information for Promotion Tools

The reference information in this chapter applies to the following promotion tools:

- the Export SAS Package and Import SAS Package wizards in SAS Management Console, SAS Data Integration Studio, and SAS OLAP Cube Studio

- the batch export and import tools

The reference information includes:

- details about promoting specific object types
- the types of physical content that are promoted along with metadata
- considerations for promoting metadata from previous versions of SAS
- troubleshooting information.

Promotion Details for Specific Object Types

Promoting Access Controls

About Promoting Access Controls

When you promote objects, you can choose to also promote the objects' access controls, subject to the following limitations:

- In order for an object's ACT association to be promoted, an ACT of the same name must exist in the target metadata server. For best results, you should always promote security objects (including users, user groups, roles, and ACTs) first, in a separate package, before you promote the objects with which they are associated.

- An object's access control entries are not promoted if they refer to users or groups that do not exist in the target metadata server.

- Access control entries and ACT associations that are inherited from a folder are not promoted. For inheritance to take effect in the target metadata server, you must also promote the folder that contains the direct access control entries or ACT association.

Using the Include Access Controls Option

If you want to include direct assignments of access control entries (ACEs) and access control templates (ACTs) in the promotion process, the following steps are recommended:

- Ensure that you have done the following in the source environment:

  - Set up user groups that mirror the groups that will exist in the target environment, and define a small number of users in each group for testing purposes.

  - Create ACTs that define the permission levels for each user group, apply the ACTs to objects and folders, and test the ACTs to ensure that the appropriate level of security is applied.

  - Instead of (or in addition to) ACTs, apply ACEs to folders and objects to define the permission levels for each user group. Test the ACEs to ensure that the appropriate level of security is applied.

- Promote the ACTs and user groups to the target environment (or create them in the target environment) before you import the affected objects. Make sure that the ACTs and groups have the same names in both environments.

- When you import the affected objects to the target system, select the **Include access controls** option in the import wizard (or specify `-includeACL` in the batch import command). Each object's access control template (ACT) associations and access control entries (ACEs) will then be promoted and will be re-established in the target environment.

Note:

- In order for the ACT associations and the ACEs to be promoted, you must promote the objects or folders to which the ACTs were directly applied. Inherited ACT associations and ACEs are not promoted.

- If the imported objects already exist on the target metadata server, then any existing access controls for these objects (including access controls that are inherited) will be overwritten by the imported ACTs and ACEs.

- Access controls for folders are promoted only if the imported folders do not already exist on the target metadata server. If a folder already exists, its access controls are not overwritten by the imported ACTs and ACEs.

- Starting in SAS 9.4, the import process automatically grants the importing user WriteMetadata permission on newly imported objects if the user does not already have this permission.

Applying Access Controls Directly in the Target Environment

Instead of promoting access controls, you can choose to apply ACTs or ACEs to folders and objects directly in the target environment. For this option, you need to define user groups, ACTs, and ACEs only in the target environment instead of in both environments. This option involves fewer steps, but it does not provide the opportunity to test your security settings before promotion.

For this option, do not select the **Include access controls** option in the import wizard. If you are using the batch import command, then omit the **-includeACL** option. Access controls for the imported objects will be applied as follows:

- If you import an object that already exists in the target metadata server, then the permissions that have been applied to the existing object are preserved when you overwrite the object.

- If you import an object for the first time, then the object inherits permissions from the folder into which the object is imported.

Promoting Dashboard Objects

Dashboard objects (including dashboards, indicators, indicator data, ranges, indicator configurations, and dashboard components) can be promoted from a SAS 9.3 (SAS BI Dashboard 4.31) deployment to a SAS 9.4 (SAS BI Dashboard 4.4) deployment.

However, these object types cannot be promoted from a SAS 9.2 (SAS BI Dashboard 4.3) deployment to a SAS 9.4 (SAS BI Dashboard 4.4) deployment. You must use the SAS Migration Utility instead.

Promoting Libraries and Tables

The following special considerations apply to promoting libraries and tables:

- Library names must be unique within a SAS Application Server. When you import a library, the promotion tools check to determine whether the target environment contains a library that has the same name and is located on the same application server. If such a library exists, then the promotion tool renames the imported library to *library-name*(*n*), where *n* is an integer value.

- Table names must be unique within a library. When you import a table, the promotion tools check to determine whether the target environment contains a table

that has the same name and is part of the same library. If such a table exists, then the promotion tool renames the imported table to *table-name*(*n*).

- The promotion tools do not create the physical directory paths that are needed for libraries. Instead, the import tools check to determine whether the directory that is defined in metadata exists in the target environment.

- If you choose to export physical tables along with metadata, you should be aware that large data sets could require lengthy processing times (since data sets are packaged in the SPK file along with the metadata).

  Note: When you use export table objects with the batch export tool, the physical tables cannot be exported.

- If you import metadata for a table or an external file that is already registered in the target environment, then any matching column metadata is overwritten by default. Any non-matching metadata in the target table or file is preserved. The matching of column names is case-insensitive. For example, metadata for a column called **SALES** will overwrite metadata for a column called **Sales**.

  To override the default behavior for resolving column differences, you can use the Change Analysis feature of the import wizard or set global options for the batch import tool.

- If you export a SAHDAT library that has encryption properties set, the encryption key is not included in the export.

Promoting OLAP Content

An environment can contain only one OLAP schema for each SAS Application Server. If an application server has an OLAP schema and you try to import another one for the same server, the import wizard prompts you to either deselect the schema or select a different application server.

The following special considerations apply to promoting OLAP cubes:

- Before you promote cubes, you must create directories on the target system for the associated physical files (including cube header files, MOLAP aggregation tables, drill-through tables, and ROLAP aggregation tables).

- OLAP cubes must be promoted along with their associated jobs. The export wizard will not allow you to continue if you do not select both objects. You can use the **Dependencies** tab to identify the jobs.

- Cubes and their associated jobs must be unique within a SAS Application Server. When you import one of these objects, the promotion tools check to determine whether the target environment contains a cube or job that has the same name and is located on the same application server. If such a cube or job exists, then the promotion tool renames the imported cube or job to *object-name*(*n*), where *n* is an integer value.

- After you promote the metadata and jobs for an OLAP cube, you must rebuild the cube.

For important details about these and other tasks that are associated with promoting cubes, see the *SAS OLAP Server: User's Guide*.

Promoting Portal Content

The wizards and batch tools cannot promote portal pages, page templates, portlet instances, or the portal's application and user permissions trees. To promote these items, you can use the content promotion tool for the SAS Information Delivery Portal 4.4. See "About the Portal Promotion Tools" in *SAS Intelligence Platform: Web Application Administration Guide*.

Promoting Prompts

If you promote an object (for example, a stored process or a shared prompt) that includes a **File or directory** prompt, then you must do the following:

• After the promotion is complete, you must edit the prompt to ensure that a valid workspace server is specified.

• If the **File or directory** prompt has a default file or directory value specified, then you must move the associated physical file or directory to the target environment, if it is not already present. After the promotion, you must edit the prompt to associate it with the correct file or directory.

Promoting Repositories

Objects that are checked out from the project repository cannot be exported with the promotion tools. In order to export these objects, you must first check them into the Foundation or Custom repositories.

For more information, see "About SAS Metadata Repositories" on page 227 .

Promoting Queue Managers

If you export a queue manager for WebSphere object or a queue manager for MSMQ object, then all of the message queues that are defined for that object are also exported, regardless of whether they were selected. When you import the package, the message queues might be added to your metadata. If you do not want to use the message queues, you should delete them when the import is complete.

Promoting SAS Enterprise Guide Projects and SAS Add-In for Microsoft Office Documents

After you promote a SAS Enterprise Guide project or a SAS Add-In for Microsoft Office document, you might need to update the application server names for tables that are associated with the project or document. This step is necessary if the target environment uses different application server names than the source environment. After you perform the promotion, use one of the following methods in the target environment to adjust the metadata:

• Run the migration wizard that is provided with SAS Enterprise Guide and SAS Add-In for Microsoft Office. For details, see "Administering SAS Enterprise Guide" in *SAS Intelligence Platform: Desktop Application Administration Guide*; or see "Migrating SAS Content" in the SAS Add-In for Microsoft Office Help.

• For SAS Enterprise Guide projects, use the Project Maintenance dialog box to edit the table metadata. For details, see the SAS Enterprise Guide Help.

Promoting Secured Data Folders, Secured Library Objects, and Secured Table Objects

The following special considerations apply to promoting secured data folders, secured library objects, and secured table objects that have been created to support metadata-bound libraries:

- The target metadata folder path and folder name must be the same as the source metadata folder path and folder name. Import and export of secured data folders, secured library objects, and secured table objects is primarily for transfer of these metadata objects from one metadata server to another. For example, you might promote these objects from a development server to a test server and then to a production server.

  Note: It is also possible to import these objects into the same metadata server, folder path, and folder from which you exported them. You might use this method to back up and restore these objects.

- You cannot explicitly choose which secured table objects to import or export. All secured table objects within each secured library object are ported with that secured library object.

- After the initial import of a secured library object into a new target metadata server, you must use SAS code to apply passwords to the new target library. (For security reasons, passwords are not ported.) See documentation for the MODIFY statement in the *SAS Guide to Metadata-Bound Libraries*.

  T I P When you submit the code to add password information to the secured library object in the new target server, remember to specify options (or adjust configuration) in your SAS session so that it connects to the new target metadata server (where the passwords do not yet exist), instead of the source metadata server (where the passwords already exist).

  This requirement applies to only the first import of each secured library object into a new server (and to the first such import after the passwords for the source library object are changed). Other imports of the same secured library object into the same target server include a step that retrieves the already-present passwords from the pre-import target objects and updates the post-import target objects with those passwords.

- As a best practice, you should maintain a separate copy of your physical data for each environment (for example, development, test, and production). The alternative, pointing multiple metadata servers at the same physical data, is supported but introduces significantly more complexity.

Note: Like promotion of a regular folder, promotion of a secured data folder can include direct access controls (both explicit and ACT) only if the corresponding secured data folder does not already exist in the target server. If the corresponding secured data folder already exists in the target server, that folder's direct access controls are not modified, updated, or overwritten by the import process.

For information about metadata-bound libraries, see the *SAS Guide to Metadata-Bound Libraries*.

Promoting Security Objects and Server Objects

About Promoting Security Objects and Server Objects

You can promote security objects (users, user groups, roles, and access control templates) and all of the server types that can be registered in metadata. The promotion of these objects can be useful when you are creating or refreshing a SAS environment that mirrors an existing environment. For example, you can use this feature to help to create a test environment that mirrors your production environment and to refresh the test environment as needed. Or you might choose to create server or security objects in a test environment and then promote the objects to your production environment after the objects have been tested.

To support the promotion of server and security objects, the SAS Folders tree in SAS Management Console contains virtual folders called **Servers** and **Security**. These folders are referred to as virtual folders because they do not actually store metadata

objects. The virtual folders are represented by white folder icons ( ) and are located in the path **/System** in the Foundation repository and in each custom repository. These folders appear only in SAS Management Console. Therefore, you can promote server and security objects only with SAS Management Console and the batch export and import tools. You cannot use SAS Data Integration Studio or SAS OLAP Cube Studio to promote these objects.

Note: Server and security objects can reside only in the appropriate virtual folders. They cannot be moved or promoted to other folders. In addition, the **Server** and **Security** virtual folders can be used only for server and security objects. You cannot move or promote other types of objects into these folders.

Follow these guidelines when importing security and server objects:

- If a package includes security objects and server objects, then you must import the package to either the root folder (**SASFolders**) or to **/System**. When importing these objects to the root folder, be sure to select the option **Preserve source path information for objects in the package** so that the wizard will place each object in the appropriate virtual folder.

- If the package contains additional objects that are not security or server objects (for example, if the package contains stored processes), then you must import the package at the **SASFolders** level. Be sure to select the option **Preserve source path information for objects in the package** so that the wizard will place each security and server object in the appropriate virtual folder.

CAUTION:
For best results, you should always promote security or server objects first, in a separate package, before you promote the objects with which they are associated.

Promoting Security Objects (Users, User Groups, Roles, and Access Control Templates)

You can promote security objects from the **Security** virtual folder, which is located in the path **/System**. The folder has subfolders for access control templates, authentication domains, roles, user groups, and users, as shown here:

```
⊟ ☐ Security
    ┈☐ Access Control Templates
    ┈☐ Authentication Domains
    ┈☐ Roles
    ┈☐ User Groups
    ┈☐ Users
```

Note:

- Only unrestricted users are able to import access control templates.

- Authentication domains cannot be promoted. If an imported user or server depends on an authentication domain that does not exist in the target environment, the domain is automatically created during the import process.

- The **Authentication Domains** and **Users** subfolders are present only in the Foundation repository.

- Passwords are not promoted and must be added manually after the security objects are promoted. Recommended methods for adding and updating passwords include the following:

- SAS Management Console

  See SAS 9.4 Management Console: Guide to Users and Permissions for more information.

- Personal Login Manager

- Login Manager within SAS Enterprise Guide

- SetPassword utility, which is located in **SASHome/ SASPlatformObjectFramework/9.4**

The following special considerations apply to the promotion of security objects:

- Associations among objects are retained as follows:

  - When you import objects (for example, information maps) whose security is being controlled, their associations with security objects (users, user groups, and ACTs) are retained if the security objects already exist on the target system or are promoted in the same package. As a best practice, you should import security objects first in a separate package.

  - When users are imported, their associations with ACTs, user groups, and roles are retained if these objects either exist in the target system or are promoted in the same package as the users.

  - When user groups are imported, their associations with ACTs, users, and roles are retained if these objects exist in the target system or are promoted in the same package as the groups.

  - When roles are imported, their associations with users and user groups are retained if the users and user groups exist on the target system or are promoted in the same package as the roles.

  - When ACTs are imported, their associations with users and user groups are retained if the users and user groups exist on the target system or are promoted in the same package as the ACTs.

  If the associated objects do not exist in the target system (or are not included in the package) as described in the preceding list, the associations are removed.

  On the Select Objects to Export page, you can use the **Dependencies** and **Used by** tabs as needed to identify and select the groups that a user belongs to, the users that

make up a group, and the users and groups that an ACT depends on. (Dependency and "used by" information is not available for roles.)

- If a package includes multiple types of security objects, then you must import the package to **/System/Security** or higher. If you import the package to the root folder (**SASFolders**), be sure to select the option **Preserve source path information for objects in the package**.

- When users are imported, the following special considerations apply to the promotion of user login (account) information:

    - If a user already exists on the target system, the import operation does not overwrite the user's existing logins. However, if an imported user object contains a login for an authentication domain for which the user has no existing logins, the login is added.

        For example, suppose the target system contains one login for a user named Joel, and that the login specifies the DefaultAuth domain. If the imported package contains a login for Joel that specifies the DefaultAuth authentication domain, then that login is not imported. If the imported package contains a login for Joel that specifies a different domain (for example, Auth2), then that login is imported.

    - Passwords that are included in the logins of external accounts are not included in the import. These passwords must be added to logins manually after the import is complete.

        Note: Passwords for internal accounts are preserved.

    - Effective with SAS 9.4, machine names in stored user IDs are updated to the target metadata server machine name during the import process.

- Users cannot be promoted to a custom repository.

- When a user is promoted, application preferences that are associated with the user are not included in the promotion. (However, preference information is included in migrations that are performed using the SAS Migration Utility. See the *SAS Intelligence Platform: Migration Guide*.)

- When a user object is overwritten by an imported user object, any application preferences that are associated with the user on the target system are erased.

Promoting Server Objects

You can promote server objects from the **Server** virtual folder, which is located in the path **/System**. SAS 9.4 supports the promotion of all of the server types that are registered in metadata.

If a server that you want to promote does not appear in the **Server** folder after you upgrade to SAS 9.4, you need to run the Upgrade Metadata utility. Follow these steps:

1. Back up your metadata server. See Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169.

2. Run the Upgrade Metadata utility. To do so, go to the **Plug-ins** tab of SAS Management Console and select **Environment Management** ⇨ **Metadata Manager** ⇨ **Active Server**. Then right-click the **Active Server** node, and select **Upgrade Metadata**.

    After the utility executes, the servers should appear in the **System/Servers** folder.

The following special considerations apply to the promotion of servers:

- Server names must be unique within a metadata server. If the target metadata server already has a server with the same name as an imported object, then the promotion tool renames the imported object to *object-name*(*n*), where *n* is an integer value.

- The import wizard displays prompts that enable you to provide new values for host machine names, server ports, and paths and script names for server start-up commands. For each of these attributes, provide values that are valid for the target environment. The Summary page that appears at the end of the wizard process displays the old and new values for each attribute. Before completing the import operation, review this page to ensure that your entries are correct.

   If you are using the batch import tool, then use the substitution properties file to specify these values.

   Note: When you promote a SAS/SHARE server, the server's port number cannot be updated.

- The promotion tools do not copy the physical files that are associated with servers, such as start-up scripts and server configuration files. These files must be copied to the target environment manually.

If you need to promote a SAS Application Server and its components, see the additional considerations in "Promoting SAS Application Servers and Associated Spawners" on page 300.

Promoting SAS Application Servers and Associated Spawners

Generally, when you promote objects that depend on SAS Application Servers, it is not necessary to promote the application server objects. Instead, you can use the import wizard (or the substitution properties file) to associate imported objects with servers that already exist on the target system.

For situations in which you do need to promote metadata for application server objects, the **Servers** virtual folder contains a node for each SAS Application Server and each application server component that is configured on your system, as shown here:
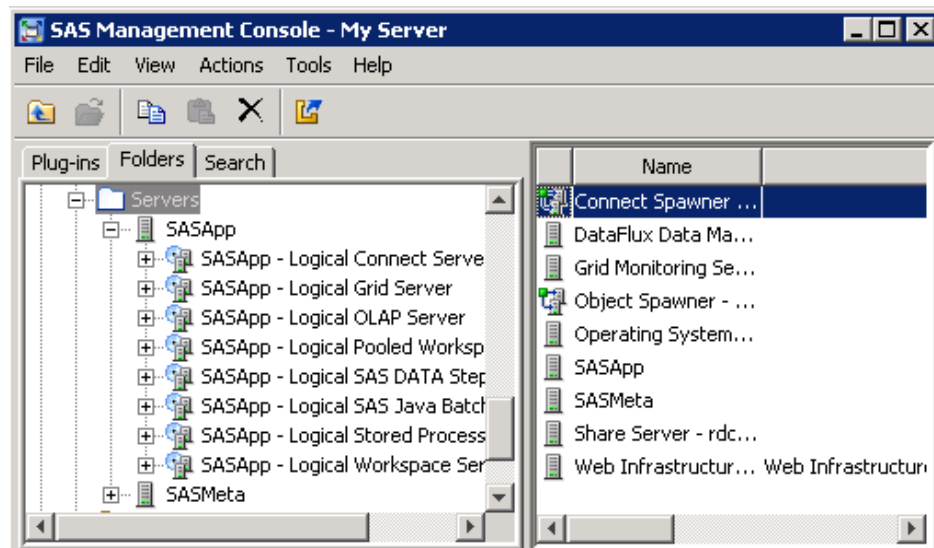
The spawners that are associated with the servers are displayed in the right panel when the **Servers** folder is selected. The spawners do not appear on the **Folders** tab.

The following special considerations apply to the promotion of SAS Application Servers:

- When you promote an application server, all of its components are included in the promotion. You cannot select individual server components (for example, specific workspace servers) to export or import.

- On the Select Objects to Export page, you can use the **Dependencies** tab as needed to identify and select the spawners that are associated with each application server.

Promoting SAS Visual Analytics Content

Promoting SAS Visual Analytics Collections Folders

If you promote SAS Visual Analytics collections folders, the object references that the folders contain (for example, references to reports and explorations) are promoted. The objects themselves (that is, the referenced reports and explorations) are not included in the promotion.

To make the objects available, promote them to the same folder location in the target environment where are they are stored in the source environment.

Promoting SAS Visual Analytics Reports

Consider the following when promoting SAS Visual Analytics reports:

- For any report object that has a legend in SAS Visual Analytics 7.5, the **Automatic** setting is not available for the **Visibility** option. If you open a report created in SAS Visual Analytics 7.4 for which the **Visibility** option is set to automatic, you will see that option displayed in SAS Visual Analytics 7.5. You can preserve the **Automatic** setting. However, if you want to change the **Visibility** option, you can specify only that the legend is displayed (**On**) or hidden (**Off**).

- When opening a newly promoted report with SAS Visual Analytics Viewer, the decision about how to render axes is based on the **Measure layout** parameter rather than the data. Therefore, you might get a different rendering of the axes and legends than what was expected from the original report. However, when the promoted report is opened with SAS Visual Analytics Designer, the data is accessible and the rendering of axes and legends should be as expected.

- When a coordinates geo map is promoted from SAS Visual Analytics 7.4 to SAS Visual Analytics 7.5, the marker shape is not promoted correctly. As a workaround, change the map type to **Bubbles** and then back to **Coordinates**.

Types of Physical Content That Are Promoted with Metadata Objects

For most object types, the export and import wizards (and the batch export and import tools) promote both the physical content and the metadata. When you use the batch tools, the promotion automatically includes all associated physical content except tables and external files. When you use the wizards, you can choose whether to include or exclude some of the content types by using the **Options** tab.

The following table provides examples of the types of physical content that are exported and imported with metadata objects.

Table 22.1 *Promotion of Physical Content with Metadata Objects*

Object	Associated Physical Content	Promotion Details
Channel	Packages published to the channel's archive or WebDAV location	Associated physical content is always promoted with batch tools and is optionally promoted with wizards.
Cube	Cube header files, MOLAP aggregation tables, drill-through tables, and ROLAP aggregation tables	Associated physical content is not promoted. For details, see the *SAS OLAP Server: User's Guide*.
External file	Physical file	Associated physical content is not promoted with batch tools and is optionally promoted with wizards.
	External format file (EFF)	Associated physical content is not promoted with batch tools and is optionally promoted with wizards.
Job	Custom code, preprocesses, and postprocesses	Associated physical content is always promoted.
Library	XML files and XML maps that are contained in the library	Associated physical content is always promoted with batch tools and is optionally promoted with wizards.
Prompt	Physical files that are associated with **File or directory** prompts	Associated physical content is not promoted. For details, see "Promoting Prompts" on page 295.
Report	XML (.srx) file that defines the report	Associated physical content is always promoted. The target is the path on the SAS Content Server that is mapped to the target metadata folder via content mapping.
Stored process	Source code (.sas) file	Associated physical content is optionally promoted with the wizards and is always promoted with the batch tools. The target is the source code repository that you designate in the wizard dialog box. If the repository definition does not exist on the target system, the import wizard enables you to add it. When you use the batch import tool, you can use the substitution file to designate the source code repository and to specify that the repository definition is to be created if it does not exist.

Object	Associated Physical Content	Promotion Details
Table	Physical table	Associated physical content is not promoted with the batch tools and is optionally promoted with the wizards.
		Use this option with small data sets only. Since data sets are packaged in the SPK file, large data sets could require lengthy processing times.

See Also

"Promoting Physical Content along with Metadata" on page 270

Special Considerations for Promoting Content to SAS 9.4 from an Earlier SAS Version

About Promoting Content to SAS 9.4 from an Earlier SAS Version

You can export a package from a SAS 9.2 or SAS 9.3 SAS Metadata Server and import it to a SAS 9.4 Metadata Server. Using the SAS Migration Utility to migrate from SAS 9.1 to SAS 9.4 is not an option. Promotion of content might be supported, but you should contact SAS Technical Support before performing any action. Promotion of content is applicable in the following scenarios:

- You are upgrading to SAS 9.4, but you chose not to use the SAS Migration Utility. For details, see *SAS Intelligence Platform: Migration Guide*.

- You are running SAS 9.4 in parallel with SAS 9.2 or SAS 9.3, and you want to update your SAS 9.4 system with new content that you created in the earlier release.

The following tasks are important when you promote content to SAS 9.4 from an earlier version:

- correctly promote personal folders

- address special considerations for promoting folders, dashboard objects, and portal content to SAS 9.4

Note: If you need to promote content to support an upgrade from SAS 9.1.3 to SAS 9.4, then please contact SAS Technical Support for more information, and visit SAS Migration Utility.

Promoting Personal Folders from SAS 9.2 to SAS 9.4

Options for Correctly Promoting Personal Folders from SAS 9.2

In SAS 9.2, users' personal folders are located by default in a parent folder named **Users**. When you promote, the name of this folder in your SAS 9.4 deployment is **User Folders** by default.

CAUTION:

> **Be sure to follow the instructions in this section to ensure that your users' personal folders will work properly after promotion.**

You can use one of two options to correctly promote your users' personal folders.

Option 1: Promote Personal Folders to the New Location in SAS 9.4

You can promote users' personal folders from SAS 9.2 to this new location (**User Folders**) in SAS 9.4. To do so, you must promote the personal folders separately from other metadata. In addition, you must promote the personal folders without promoting the **Users** folder itself. Follow these steps:

1. Use SAS Management Console to connect to the SAS 9.2 SAS Metadata Server, and select the **Users** node in the SAS Folders tree.

2. In the right pane, select all of the personal folders that you want to export. (Press CTRL to enable the selection of multiple items.) Then, right-click, select **Export SAS Package**, and complete the export.

3. Use SAS Management Console to connect to the SAS 9.4 SAS Metadata Server. Select the **User Folders** node in the SAS Folders tree, and import the package.

Option 2: Configure SAS 9.4 to Recognize the Users Folder Name, and Promote the Existing Folder

If you do not want to switch to the new folder name (**User Folders**) in SAS 9.4 for personal folders, you can reconfigure SAS 9.4 to use the previous folder name (**Users**). For instructions, contact SAS Technical Support.

Restriction on Specifying Locations for SAS 9.2 Folders in a SAS 9.4 Custom Repository

In SAS 9.3 and SAS 9.4, the import wizard and the batch import tool enable you to reorganize content as part of the promotion process. You can do so by specifying new folder locations for individual objects or folders that you are importing.

When you are importing SAS 9.2 content to a SAS 9.4 custom repository, you cannot change the location of a folder. However, you can change the location of individual objects in the SAS 9.4 custom repository.

Promoting Dashboard Objects from SAS 9.2 or SAS 9.3 to SAS 9.4

Dashboard objects (including dashboards, indicators, indicator data, ranges, indicator configurations, and dashboard components) can be promoted from a SAS 9.3 (SAS BI Dashboard 4.31) deployment to a SAS 9.4 (SAS BI Dashboard 4.4 or later) deployment.

However, these object types cannot be promoted from a SAS 9.2 (SAS BI Dashboard 4.3) deployment to a SAS 9.4 (SAS BI Dashboard 4.4 or later) deployment. Please contact SAS Technical Support for guidance.

Promoting Portal Content from SAS 9.2 or SAS 9.3 to SAS 9.4

To promote portal pages, page templates, portlet instances, and the portal's application and user permissions trees from SAS 9.2 or SAS 9.3 to SAS 9.4, use the content promotion tool for SAS Information Delivery Portal 4.4. See "About the Portal Promotion Tools" in *SAS Intelligence Platform: Web Application Administration Guide*.

Troubleshooting the Promotion Tools

The following error messages might appear when you run the Export SAS Package wizard, the Import SAS Package wizard, or the batch import or export tool:

Solutions for Error Messages

`Out of memory error occurred during the import (or export) process.`

This message indicates that an out of memory error occurred on the client machine. The error could occur if you export or import a large number of sizable objects, such as information maps or SAS Data Integration Studio jobs. If this error occurs, you can take either of the following actions:

- Instead of exporting all of your objects at the same time, perform multiple exports that include a smaller number of objects.

- Increase the amount of memory that is allocated to the import or export process. For instructions, see "Ensure Adequate Memory" on page 272.

`ERROR - Failure exporting metadata from source repository. Reason: org.omg.CORBA.NO_MEMORY: Insufficient memory`

This message, or a similar message, appears if an out-of-memory error occurs on the metadata server machine. Solutions might include the following:

- Check the memory settings on the server machine and adjust them as necessary.

- Use the Analyze and Repair Metadata utility in SAS Management Console to clean up orphaned metadata. Then use the Reorganize Repositories option of the metadata server backup facility to reclaim unused disk space left from deleted metadata objects.

- Instead of exporting all of your objects at the same time, perform multiple exports that include a smaller number of objects.

`The export process has failed. No X11 DISPLAY variable was set, but this program performed an operation which requires it. For more information, view the export log file:` *filename*

This error could occur if you submit a batch import or export job on a UNIX machine for which the DISPLAY variable has not been set. To resolve the error, set the DISPLAY variable by entering a command such as the following:

`export DISPLAY=`*machine-name*

`No extraField1 found.`

This error can occur if you try to import a package file whose contents were updated after the package was created. For example, if you open a package file, save any of the files in the package (for example, the export log), and then save the package file, then the package becomes corrupted and cannot be imported.

To resolve the error, perform the export again to re-create the package.

The following resources could not be found in the package file
or on the target server. Objects dependent on these resources
may not function properly after the import process.

This message means that the objects that you are importing depend on objects that are not present on the target metadata server.

You will need to import the resources that objects depend on, unless those resources already exist in the target environment. For example, in order for a report to function properly, you must import the information map that the report depends on, as well as any stored processes that are associated with the report.

Destination folder has invalid or missing content mapping;
cannot import reports.

Content mapping, which is set up during the installation and configuration process, establishes the location of the XML (.srx) files that define reports. The location must be a path on the SAS Content Server. If this message appears when you try to run the import wizard, click **OK** to close the message box and click **Cancel**. Then follow these steps to check content mapping on the target metadata server:

1. Right-click the appropriate folder in the SAS Folders tree, and click **Properties**. Generally, content mapping is specified in the root folder (**SAS Folders**).

2. Open the **Content Mapping** tab, and make any necessary changes. When you specify the location on the SAS Content Server, you must enter the user ID and password for the SAS Trusted User (sastrust) in the appropriate fields. Then start the import again.

Could not import "*filename.sas*" in "*path*" on application server
"*server name*". Reason: Insufficient authorization to access *path*.

This message appears in the error log at the end of the import process. It means that the user who ran the import does not have Write permission for the specified directory in the file system of the target machine. Grant the user Write permission to the directory, and run the import again.

Libname *library name* is not assigned.

This message could appear in the error log if you try to export physical tables along with DBMS table objects while you are logged on as an unrestricted user. Unrestricted users (for example, sasadm) cannot access the servers that are necessary to export physical DBMS tables. Log on as a different user, and start the export process again.

Part 8

SAS Intelligence Platform Batch Tools: Reference

Chapter 23

Using the SAS Intelligence Platform Batch Tools

About the SAS Intelligence Platform Batch Tools

The SAS Intelligence Platform provides a variety of batch tools that you can use to perform actions on objects and other components of the SAS Intelligence Platform. The framework is provided along with applications such as SAS Management Console, SAS Data Integration Studio, SAS Information Map Studio, and SAS OLAP Cube Studio. Using the SAS Intelligence Platform batch tools, you can perform tasks independent of these applications. The batch tools can be incorporated into scripts so that you can run them repeatedly on either an ad hoc or scheduled basis.

The SAS Intelligence Platform provides the following categories of batch tools:

- metadata management tools, which you can use for tasks such as listing selected objects, deleting selected objects, creating new folders, and managing metadata access.

- export and import tools. These tools enable you to promote individual objects or groups of objects from one SAS deployment to another, or from one folder location to another within the same deployment. The promotion includes all associated content except physical files for tables and external files.

- batch relationship reporting tools, which enable you to identify relationships among the content objects in the SAS Folder tree. For example, you can identify the objects that a given object depends on or contains; the objects that depend on or contain a given object; and the objects that are associated with a given object. Both direct and nested relationships can be identified.

- metadata server administration tools, which administrators can use to perform tasks such as executing metadata server backups and restores, creating and deleting metadata repositories, and updating metadata profiles.

- the Deployment Backup and Recovery tool, which provides an integrated method for backing up and recovering your SAS content across multiple tiers and machines.

Additional batch tools are available for middle-tier administration. See "Use the SAS Web Infrastructure Platform Utilities" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

General Instructions for Using the SAS Intelligence Platform Batch Tools

Accessing the SAS Intelligence Platform Batch Tools

You can execute the SAS Intelligence Platform batch tools on any machine where the SAS Platform Object Framework is installed. The commands are located in the following paths:

- The batch export and import tools are located in the following path:

  SAS-installation-directory/SASPlatformObjectFramework/9.4/

- Batch tools for managing metadata are located in the following path:

  **SAS-installation-directory/SASPlatformObjectFramework/9.4/
  tools**

- Batch tools for administrators are located in the following path:

  **SAS-installation-directory/SASPlatformObjectFramework/9.4/
  tools/admin**

Special Instructions for Windows Systems

When using the SAS Intelligence Platform batch tools on Windows 7, Windows 8, Windows Server 2008, and Windows Server 2012 machines, you might need to open the command window with the Run as Administrator option to obtain the required level of access to directories and files. To enable this option, select **Start ⇨ All Programs ⇨ Accessories**. Then right-click **Command Prompt** and select **Run as administrator**.

Special Instructions for z/OS Systems

When using the SAS Intelligence Platform batch tools on z/OS systems, you might need to precede special characters with a backslash character (\). Characters other than letters (A–Z and a–z), numbers (0–9), underscores (_), hyphens (-), and periods (.) might require a preceding backslash.

About Case Sensitivity

In all of the SAS Intelligence Platform batch tools, you must use the correct case for option names (for example, -includeDep and -newOnly) and object types (for example, **InformationMap**).

All other elements of the commands are case insensitive.

Specifying Dates, Times, and Time Periods

Specifying an Absolute Date (or an Absolute Date and Time)

When you use -since or -before in a command, you can specify an absolute date (or an absolute date and time). Use one of the following formats:

- ddMMMyyyy

- ddMMMyyyy:HH:mm:ss

- ddMMMyy

- ddMMMyy:HH:mm:ss

- MM/dd/yyyy

- MM/dd/yyyy HH:mm:ss

- yyyyMMdd

- yyyyMMdd:HH:mm:ss

Note: If you do not specify a time, then the specified date begins at midnight (12:00:00 a.m.) and ends at 11:59:59 p.m.

Specifying a Relative Date

When you use -since or -before in a command, you can specify a date relative to the current date. To specify a relative date, use one of the following values:

Today
   is the current date, based on the date and time on the host machine for your client software (for example, SAS Management Console)

Yesterday
   is the day just before the current date.

"Current day of last year"
   is the same as the current date, except that the year is replaced with the previous year. For example, if the current date is October 12, 2013, then **"Current day of last year"** is October 12, 2012. February 29 is replaced with February 28.

"Current day of last month"
   is the same as the current date, except that the month is replaced with the previous month. For example, if the current date is October 12, 2013, then **"Current day of last month"** is September 12, 2013. If the previous month has fewer days, the date is adjusted downward as necessary. For example, if the current date is October 31, 2013, then **"Current day of last month"** is September 30, 2013.

"Current day of last week"
   is seven days previous to the current date. For example, if the current date is October 12, 2013, then **"Current day of last week"** is October 5, 2013.

"*n* days ago"
   is *n* days previous to the current date. When specifying this option, replace *n* with an integer.

Note: Dates are assumed to begin at midnight (12:00:00 a.m.) and end at 11:59:59 p.m.

Specifying a Relative Time Period

When you use -since in a command, you can specify a time period that is relative to the current date. To specify a relative time period, specify one of the following values:

"Year to date"
> is the period from January 1 of the current year up to and including the current date.

"Month to date"
> is the period from the first day of the current month up to and including the current date.

"Week to date"
> is the period from the most recent Monday up to and including the current date.

Note: Dates are assumed to begin at midnight (12:00:00 a.m.) and end at 11:59:59 p.m.

Specifying Object Types

For some of the metadata management batch tools, you might need to specify one or more object types, as follows:

- in the -objects option, if you specify individual objects. You must specify the object type in parentheses after each object name.

- in the -types option.

The following table provides a list of object type names that you can use with these options.

CAUTION:
> **Object types are case sensitive. Be sure to enter them exactly as they are shown here.**

Note: Your deployment might include additional object types, depending on which applications you have installed.

ACT	InformationMap.OLAP	Report.BI
Action	InformationMap.Relational	Report.Component
Application	JMSDestination (Java Messaging System message queue)	Report.Image
ApplicationServer	Job	Report.StoredProcess
BurstDefinition	Job.Cube	Role
Channel	Library	SearchFolder
Condition	MessageQueue	SecuredLibrary
ConditionActionSet	MiningResults	Server
ContentSubscriber	MQM.JMS (queue manager for Java Messaging Service)	Service.SoapGenerated
Cube	MQM.MSMQ (queue manager for MSMQ)	SharedDimension
DataExploration	MQM.Websphere (queue manager for WebSphere MQ)	Spawner.Connect
DeployedFlow	Note	Spawner.IOM (object spawner)
DeployedJob	OLAPSchema	StoredProcess
Document	Project	SubscriberGroup.Content

EventSubscriber	Project.EG	SubscriberGroup.Event
ExternalFile	Project.AMOExcel	Table
FavoritesFolder	Project.AMOPowerPoint	User
Folder	Project.AMOWord	UserGroup
Folder.SecuredData	Prompt	VisualDataQuery
GeneratedTransform	PromptGroup	VisualExploration
InformationMap	Report	

Note: By default, if you specify an object type that has one or more subtypes, then the subtypes are included. For example, if you specify Report, then Report.BI, Report.Component, Report.Image, and Report.StoredProcess are automatically included.

Note: For server objects, the preceding list does not include all of the various subtypes. If you want to specify a server subtype, you can use SAS Management Console to look up the subtype name. On the **Folders** tab, navigate to **/System/Types**. On the right pane, right-click an object type and select **Properties**. The Type Name is displayed on the **Advanced** tab of the Properties dialog box.

Common Options

When using most of the SAS Intelligence Platform batch tools, you must provide connection options to log on to the SAS Metadata Server. These options, which are represented in syntax statements as *connection-options*, are as follows:

Table 23.1　*Connection Options to Log On to the SAS Metadata Server*

Option	Description
-host *host-name*	Identifies the host machine for the metadata server. This option is required if the -profile option is not set.
-port *port*	Specifies the port on which the metadata server runs. This option is required if the -profile option is not set.
-user *user-ID*	Specifies the user ID of the connecting user. This option is required if the -profile option is not set.
-password *password*	Specifies the password of the connecting user. This option is required if the -profile option is not set.

Option	Description
-profile "*profile-name*"	Specifies the name of the connection profile that is to be used to connect to the metadata server. You can provide this option in place of -host, -port, -user, and -password.
	Specify any connection profile that has been created for use with client applications. The profile must exist on the computer where the command is executed. To see your available profiles, open a client application such as SAS Management Console. The connection profiles are displayed in the drop-down box in the Connection Profile dialog box. Or you can look in the following directory:
	• Windows: **C:\Users*user-name*\AppData\Roaming\SAS\MetadataServerProfiles**
	• UNIX: ***/user-home*/.SASAppData/MetadataServerProfiles**
	For example, if the directory contains a file called My Server.swa, you would specify **-profile "My Server"**. Specify the profile name only. Do not specify the filename extension or the path.

Note: You must provide connection options to log on to the SAS Web Server or SAS Web Application Server (not the metadata server) when submitting some commands, including the following:

- Most of the Deployment Backup and Recovery batch commands. See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

- The batch relationship reporting tools. See "Common Options for the Relationship Reporting Tools" on page 363.

The following additional options can be specified with any of the metadata server administration batch commands:

Table 23.2   Other Common Options

Option	Description
-log *log-path	log-path-and-filename*	Specifies the path (or the path and filename) where the log file is to be written.
-help	Displays online help for the command.

Chapter 24

Batch Tools for Metadata Management

About the Batch Tools for Metadata Management

The following batch tools are provided for metadata management:

Tool	Purpose	Documentation Location
sas-make-folder	Create a new folder in the SAS Folders tree.	"Make Folder (sas-make-folder)" on page 316
sas-list-objects	Create a filtered list of metadata objects that are stored in the SAS Folders tree.	"List Objects (sas-list-objects)" on page 317
sas-delete-objects	Delete specific objects or folders that are stored in the SAS Folders tree.	"Delete Objects (sas-delete-objects)" on page 322

Tool	Purpose	Documentation Location
sas-localize-tables	Create or modify localization metadata for data tables.	"Using the Localize Table Batch Tool" in *SAS Intelligence Platform: Data Administration Guide*
sas-relationship-reporter	Create reports about the relationships among content objects in the SAS Folders tree (for example, lineage reports and impact reports).	Chapter 26, "Using the Batch Relationship Reporting Tools," on page 347
sas-show-metadata-access, sas-set-metadata-access, and sas-make-act	Manage metadata-layer access controls.	"Batch Tools for Metadata Authorization" in *SAS Intelligence Platform: Security Administration Guide*

These tools are located in **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools**. Before using the tools, refer to the following topics:

- "General Instructions for Using the SAS Intelligence Platform Batch Tools" on page 310

- "Common Options" on page 313

Make Folder (sas-make-folder)

About the Make Folder Batch Tool

Use the Make Folder batch tool (sas-make-folder) to create a new folder in the SAS Folders tree. You can also create intermediary folders that are part of a folder's path.

Syntax for the Make Folder Tool

sas-make-folder *connection-options folder-path* <-makeFullPath>

Syntax Description for the Make Folder Tool

connection-options
See "Common Options" on page 313.

folder-path
specifies the path to the folder that is to be created. The path should be relative to the **SAS Folders** node.

By default, the command creates only one folder, and any intermediary folders in the path must already exist. If you want intermediary folders to be created, specify -makeFullPath.

-makeFullPath
creates any intermediary folders that are included in the path, if they do not already exist.

-desc
>> specifies optional descriptive text for the folder.

Examples for the Make Folders Tool

The following command creates a folder called **Orion Star** in the SAS Demo Users's personal folder. If the intermediary folder **Test Data** does not exist, it is created.

```
sas-make-folder –profile "My Server" "/User Folders/sasdemo/My Folder/
Test Data/Orion Star" -makeFullPath
```

List Objects (sas-list-objects)

About the List Objects Batch Tool

Use the List Objects batch tool (sas-list-objects) to create a list of metadata objects that are stored in the SAS Folders tree. You can filter the list based on criteria such as object name, object type, folder location, creation date and time, modification date and time, keywords, notes, and responsible user. You can create the list in text, comma-separated values (CSV), or XML format.

Syntax for the List Objects Tool

sas-list-objects

>> *connection-options*
>> <-types *object-type-1 object-type-2....object-type-n*>
>> <-folder *folder-path*>
>> <-folderTree *folder-path*>
>> <-name *character-string*>
>> <-nameMatchType CONTAINS | EQUALS | STARTSWITH>
>> <-includeDesc>
>> <-keywords *keyword*>
>> <-notes *character-string*>
>> <-extName *extended-attribute-name*>
>> <-extValue *extended-attribute-value*>
>> <-respName *name-of-responsible-user*>
>> <-respRole OWNER | ADMINISTRATOR>
>> <-since *from-absolute-date | from-relative-date|relative-time-period*>
>> <-before *to-absolute-date | to-relative-date*>
>> <-created>
>> <-modified>

Syntax Description for the List Objects Tool

connection-options
>> See "Common Options" on page 313.

-types *object-type-1 object-type-2....object-type-n*
>> specifies a list of the types of objects to be listed. Separate the object types in the list by a space. For a list of object types, see "Specifying Object Types" on page 312. Object types are case sensitive.

By default, if you specify an object type that has one or more subtypes, then the subtypes are included in the list. For example, if you specify **Report**, then **Report.Component**, **Report.Image**, and **Report.StoredProcess** are automatically included.

-name *character-string*
specifies a character string that must be present in an object or folder's name in order for it to be listed.

This option is not case sensitive. For example, if you specify **-name product**, then the tool lists objects with the names **Sales by Product** and **PRODUCT_DIM**.

If the string contains spaces, then use quotation marks.

Interactions	If you specify -includeDesc along with -name, then the object is listed if the text string is present in either the object's name or its description.
	You can use the -nameMatchType option to further define how the -name option is to be applied.

-folder *folder-path*
specifies the path to the folder to list objects from.

Interaction	This option is ignored if the -folderTree option is specified.

-folderTree *folder-path*
specifies the path to the folder and its subfolders to list objects from.

Interaction	If the -folder and -folderTree options are both specified, the -folderTree option is used.

-nameMatchType CONTAINS | EQUALS | STARTSWITH
specifies how the -name option is to be applied. The following values are valid:

CONTAINS	specifies that the object's name must contain the character string that is specified in -name.
EQUALS	specifies that object's name must exactly match the character string that is specified in -name.
STARTSWITH	specifies that the object's name must begin with the character string that is specified in -name.

The matching of character strings is not case sensitive.

Default	CONTAINS

Interactions	This option is ignored if -name is not specified.
	This option applies only to the object name. If you specify -includeDesc, this option does not apply to the description.

-includeDesc
specifies that the character string that is specified in -name is searched for in both the name and the description of each object.

Interactions	This option is ignored if -name is not specified.

The -nameMatchType option applies only to the object name. It does not apply to the object description. For the object description, a match type of CONTAINS is assumed.

-keywords *keyword*
> specifies one or more keywords to be used as filtering criteria. If an object has a keyword that exactly matches one or more of the specified keywords, then the object is included in the list. If a keyword contains spaces, then put it in quotation marks.

-notes *character-string*
> specifies one or more character strings to be used as filtering criteria for public and private Notes. If an object's public or private Notes field contains one or more of the specified strings, then the object is included in the list. If a string contains spaces, then use quotation marks.
>
> Only certain types of objects (for example, objects used by SAS Data Integration Studio) support the Notes field.

-extName *extended-attribute-name*
> specifies the name of an extended attribute. Objects that have an extended attribute that exactly matches the specified attribute name are included in the list.
>
> Only certain types of objects (for example, objects used by SAS Data Integration Studio) support extended attributes.
>
> **Interaction** You can use -extValue to specify a particular value for the attribute.

-extValue *extended-attribute-value*
> specifies a value for the extended attribute that is specified in -extName. Objects that have a value for this attribute that exactly matches the specified value are included in the list.
>
> **Interaction** This option is ignored if -extName is not specified.

-respName *name-of-responsible-user*
> specifies a user name that must be present in an object's Responsibilities metadata in order for the object to be listed. The specified name must exactly match the user's Name field in the metadata.
>
> **Interaction** You can use -respRole to specify whether the user is to be an owner or an administrator for the object.

-respRole OWNER | ADMINISTRATOR
> specifies the role of the user that is specified in -respName. The valid values are OWNER and ADMINISTRATOR. Objects are listed only if the object's Responsibilities metadata specifies this role for the user that you specify in -respName.
>
> **Interaction** This option is ignored if -respName is not specified.

-since *from-absolute-date | from-relative-date | relative-time-period*
> specifies that objects are to be listed only if they were created or modified on or after the specified date, or during the specified time period. You can specify one of the following:

from-absolute-date	specifies an absolute date.
from-relative-date	specifies a date relative to the current date.
relative-time-period	specifies a time period relative to the current date.

Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
See	"Specifying Dates, Times, and Time Periods" on page 311

-before *from-absolute-date | from-relative-date*
  specifies that objects are to be listed only if they were created or modified on or before the specified date. You can specify one of the following:

from-absolute-date	specifies an absolute date.
from-relative-date	specifies a date relative to the current date.

Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
Tip	To specify a date range, specify -since along with this option.
See	"Specifying Dates, Times, and Time Periods" on page 311

-created
  specifies that the -since and -before options are to be based on each object's Created date. If -created is specified, then objects are listed only if the Created date meets the criteria specified in -since and -before.

Default	If -since or -before is specified but -created is not specified, then -modified is assumed.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-modified
  specifies that the -since and -before options are to be based on each object's Modified date. If -modified is specified, then objects are listed only if the Modified date meets the criteria specified in -since and -before.

  Note: If an object has never been modified, then the object's Modified date is equal to its Created date.

Default	If -since or -before is specified, and if you specify neither -created nor -modified, then -modified is assumed.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-format LIST | CSV | XML
  specifies the format in which the list is to be created. The list is displayed in the command window in the selected format. No output file is created.

Example: List Format

The following command produces a generic list of all folders and objects that are contained in a particular folder in the SAS Folders tree:

```
sas-list-objects -profile "My Server" -folderTree "Shared Data/Orion Star"
```

Here is an example of the output from this command:

```
/Shared Data/Orion Star/Data(Folder)
/Shared Data/Orion Star/Reports(Folder)
/Shared Data/Orion Star/Data/Customer Orders(InformationMap)
/Shared Data/Orion Star/Data/CUSTOMER_DIM(Table)
/Shared Data/Orion Star/Data/ORDER_FACT(Table)
/Shared Data/Orion Star/Data/Orion Library(Library)
/Shared Data/Orion Star/Data/Product Sales(InformationMap)
/Shared Data/Orion Star/Data/PRODUCT_DIM(Table)
/Shared Data/Orion Star/Reports/Sales Analysis(StoredProcess)
/Shared Data/Orion Star/Reports/Sales by Customer Type.srx(Report)
/Shared Data/Orion Star/Reports/Sales by Product Group.srx(Report)
11 objects found.
```

Example: CSV Format

The following command uses a CSV-formatted list of all of the objects in the designated folder that were created before February 28, 2013.

```
sas-list-objects -profile "My Server" -folder "Shared Data/Orion Star/Reports"
-created -before 28Feb2013 -format csv
```

Here is an example of the output from this command:

```
Path,Name,TypeName,Type,Description,Created,Modified
/Shared Data/Orion Star/Reports/Sales Analysis(StoredProcess),Sales Analysis,
StoredProcess,Stored process,,"Feb 21, 2013 3:26:37 PM","Feb 21, 2013 3:26:37 PM"
/Shared Data/Orion Star/Reports/Monthly Sales.srx(Report),Monthly Sales.srx,
Report,SAS report,,"Feb 21, 2013 3:26:37 PM","Feb 21, 2013 3:26:41 PM"
/Shared Data/Orion Star/Reports/Yearly Sales.srx(Report),Yearly Sales.srx,
Report,SAS report,,"Feb 21, 2013 3:26:37 PM","Feb 21, 2013 3:26:45 PM"
```

Example: XML Format

The following command uses an XML format to list all of the objects in the designated folder that have been modified since the beginning of the current month.

```
sas-list-objects -profile "My Server" -folder "Shared Data/Orion Star/Reports"
-since "Month to Date" -format xml
```

Here is an example of output from this command:

```
<?xml version='1.0' encoding='UTF-8'?>
<Objects>
  <Object Path="/Shared Data/Orion Star/Reports/Sales Analysis(StoredProcess)"
          Name="Sales Analysis" Type="StoredProcess" TypeLabel="Stored process"
          Description="" Created="2013-02-21T20:26:37Z" Modified=
          "2013-02-21T20:26:37Z"/>
  <Object Path="/Shared Data/Orion Star/Reports/Monthly Sales.srx(Report)"
          Name="Monthly Sales.srx" Type="Report" TypeLabel="SAS report"
          Description="" Created="2013-02-21T20:26:37Z" Modified=
          "2013-02-21T20:26:41Z">
    <Responsibility Role="owner" User="SAS Demo User"/>
  </Object>
  <Object Path="/Shared Data/Orion Star/Reports/Yearly Sales(Report)"
          Name="Yearly Sales.srx" Type="Report" TypeLabel="SAS report" Description=""
          Created="2013-02-21T20:26:37Z" Modified="2013-02-21T20:26:45Z">
    <Responsibility Role="owner" User="SAS Demo User"/>
```

```
            </Object>
        </Objects>
```

Delete Objects (sas-delete-objects)

About the Delete Objects Batch Tool

Use the Delete Objects batch tool (sas-delete-objects) to delete specific objects or folders that are stored in the SAS Folders tree.

Syntax for the Delete Objects Tool

sas-delete-objects *connection-options object-path–1 object-path–2....object-path–n* <-deleteContents>

Syntax Description for the Delete Objects Tool

connection-options
specifies connection options for the metadata server from which the objects are being deleted. See "Common Options" on page 313.

object-path-1 object-path-2....object-path-n
specifies paths to folders or specific objects that are to be deleted. You can specify any number of paths. Leave a space between each path. If a path includes spaces, then enclose it in quotation marks.

Use the following syntax to specify an object path:

/folder-1/folder-2/...folder-n/<object-name(object-type)>

The following rules apply to specifying paths:

- Paths are relative to the **SAS Folders** node. Therefore, the first folder that you specify in a path must be located directly beneath **SAS Folders**.

- If you specify an object other than a folder, then you must specify the object's type. For a list of object types, see "Specifying Object Types" on page 312. Object types are case sensitive.

- If you do not specify -deleteContents, then any non-empty folders that you specify are ignored. To recursively delete a folder and all of its contents, specify the folder path without any object names, and specify -deleteContents.

- To delete objects from your personal folder, you must specify the actual path. For example, specify **/User Folders/***user-name***/My Folder**. Do not specify the shortcut (**/My Folder**).

-deleteContents
recursively deletes the contents of any non-empty folders that you specify.

Examples for the Delete Objects Tool

Here are examples of the use of sas-delete-objects:

- The following command deletes the folder called **Test Data** in the SAS Demo Users's personal folder. It also recursively deletes all objects and folders that are contained in the **Test Data** folder.

```
sas-delete-objects -profile "My Server"
"/User Folders/sasdemo/My Folder/Test Data"
-deleteContents
```

- The following command deletes the folder called **Test Data** only if it is empty. If the **Test Data** folder contains objects or other folders, then the command is ignored.

```
sas-delete-objects -profile "My Server"
"/User Folders/sasdemo/My Folder/Test Data"
```

- The following command deletes the Customer Orders information map, the CUSTOMER_DIM table, and the ORDER_FACT table, all of which are located in **/Shared Data/Orion Star Data**; and the Sales by Customer Type report, which is located in **/Shared Reports/Orion Star Reports**:

```
sas-delete-objects -profile "My Server" "/Shared Data/Orion Star Data/
Customer Orders(InformationMap)" "/Shared Data/Orion Star Data/
CUSTOMER_DIM(Table)" "/Shared Data/Orion Star Data/
ORDER_FACT(Table)" "/Shared Reports/Orion Star Reports/
Sales by Customer Type(Report)"
```

Chapter 25

Using the Batch Export and Import Tools

About the Batch Export and Import Tools

The batch import and export tools enable you to perform promotions from an operating system command line or from a batch script. These tools provide the same capabilities as the Export SAS Package and Import SAS Package wizards, with the following exceptions:

- The wizards provide the option of either including or excluding associated physical content such as table data and source code files. When you use the batch tools, the promotion automatically includes all associated content except physical files for tables and external files. For tables and external files, the batch tools do not provide the option of promoting physical files. For stored processes, you can use a global option to prevent the batch import tool from importing source code files.

- In the export wizard, you can display a list of the objects on which an object depends and then individually specify which of these objects to export. In the batch export tool, you can select a global option to include all dependent objects.

The batch export and import tools are useful for the following purposes:

- to perform ad hoc promotion jobs from the command line, as an alternative to using the wizards

- to create promotion jobs that are to be run on a scheduled basis or a repeated basis

This chapter tells how to access the batch export and import tools and provides detailed syntax descriptions and examples. As you use the batch tools, be sure to refer to the following topics:

- Chapter 20, "Promotion Tools Overview," on page 265

- Chapter 22, "Reference Information for Promotion Tools," on page 291.

Preparing to Use the Batch Export Tool or the Batch Import Tool

Perform Standard Steps to Prepare for Promotion

The batch export tool and import tool require the same preparation steps as the Export SAS Package and Import SAS Package wizards. See "Preparing to Use the Promotion Tools" on page 271. Make sure that the user who is specified in the options has the appropriate credentials. See "Ensure Appropriate User Credentials" on page 272.

Set the DISPLAY Variable (UNIX Only)

If you are using the batch import or export tool on a UNIX machine, make sure that the DISPLAY variable has been set on the machine. To set the variable, enter a command such as the following:

export DISPLAY=*machine-name*

TIP You can use the -disableX11 option to run the batch import or export tool on UNIX without setting the DISPLAY variable. This option is available with SAS 9.4M3.

Set Global Options for the Batch Export and Import Tools

Editing the .ini Files to Set Global Options
In the .ini files for the batch export and import tools, you can set global options to override the default handling of certain object types. To set these options:

1. Go to the directory **SAS-installation-directory/ SASPlatformObjectFramework/9.4** and locate the files ExportPackage.ini and ImportPackage.ini.

2. Make a backup copy of the .ini file that you want to modify.

3. Open the .ini file in a text editor, and add a new Java argument for each option that you want to specify. For example:

```
JavaArgs_10=-Dsas.promotion.stp.includesourcecode=false
JavaArgs_11=-Dsas.promotion.columns.excludenewcolumns=true
JavaArgs_12=-Dsas.promotion.columns.deletetargetcolumns=true
```

4. Save the file.

The available options are described in the following topics.

Global Option to Include or Exclude Source Code Files for Stored Processes

The following global option applies to exporting or importing stored processes using the batch tools. You can set this option in both ExportPackage.ini and ImportPackage.ini.

-Dsas.promotion.stp.includesourcecode
specifies whether source code files are to be included when a stored process is exported or imported.

- Specify **-Dsas.promotion.stp.includesourcecode=false** if you want source code files to be excluded when stored processes are imported or exported.

- Specify **-Dsas.promotion.stp.includesourcecode=true** (the default setting) if you want source code files to be included when stored processes are imported or exported.

Global Options to Resolve Column Differences in Tables and External Files

The following global options apply to importing tables and external files using the batch import tool. These options specify how column differences are to be handled if you import metadata for a table or external file and a table or external file of the same name already exists in the target folder. You can set these options in ImportPackage.ini.

-Dsas.promotion.columns.deletetargetcolumns
specifies the action that is to be taken if the target table includes columns that are not present in the imported table.

- Specify **-Dsas.promotion.columns.deletetargetcolumns=true** if you want columns to be deleted from the target table if the columns are not also present in the imported table.

- Specify **-Dsas.promotion.columns.deletetargetcolumns=false** (the default setting) if you want columns in the target table to be retained if the columns are not also present in the imported table.

-Dsas.promotion.columns.excludenewcolumns
specifies the action that is to be taken if the imported table includes columns that are not present in the target table.

- Specify **-Dsas.promotion.columns.excludenewcolumns=true** if you do not want to add any new columns to the target table.

- Specify **-Dsas.promotion.columns.excludenewcolumns=false** (the default setting) if you want columns from the imported table to be added if the columns are not present in the target table.

-Dsas.promotion.columns.excludeupdatecolumns
specifies the action that is to be taken if the imported table includes columns that are present in the target table. This option is new with SAS 9.4M2.

- Specify **-Dsas.promotion.columns.excludeupdatecolumns=true** if you do not want to import column metadata to the target table if the column already exists in the target table.

- Specify **-Dsas.promotion.columns.excludeupdatecolumns=false** (the default setting) if you want column metadata to be overwritten with column metadata from the imported table if the column already exists in the target table.

Global Option to Overwrite If Importing Pre-existing Business Rule Flow Objects

The following global option applies to importing using the batch import tool. You can set this option in ImportPackage.ini.

-Dsas.promotion.businessruleflow.overwrite

specifies whether to overwrite business rule flow objects if they already exist.

- Specify **-Dsas.promotion.businessruleflow.overwrite=false** if you do not want to overwrite existing business rule flow objects.

- Specify **-Dsas.promotion.businessruleflow.overwrite=true** (the default setting) if you do want to overwrite existing business rule flow objects.

Accessing the Batch Export and Import Tools

The batch import tool and export tools are called **ImportPackage** and **ExportPackage**. The tools are located in the following path:

***SAS-installation-directory*/SASPlatformObjectFramework/9.4**

You can execute these tools on any machine where the SAS Platform Object Framework is installed.

Using the Batch Export Tool

Syntax for the Batch Export Tool

ExportPackage

   connection-options
     -package *package-path-and-filename*
     -objects *source-location-1 source-location-2 source-location-3 ... source-location-n*
     <-types *object-type-1, object-type-2....object-type-n*>
     <-excludeSubTypes>
     <-includeDep>
     <-name *character-string*>
     <-nameMatchType CONTAINS | EQUALS | STARTSWITH>
     <-includeDesc>
     <-keywords *keyword*>
     <-notes *character-string*>
     <-extName *extended-attribute-name*>
     <-extValue *extended-attribute-value*>
     <-respName *name-of-responsible-user*>
     <-respRole OWNER | ADMINISTRATOR>
     <-since *from-absolute-date | from-relative-date|relative-time-period*>
     <-before *to-absolute-date | to-relative-date*>
     <-created>

```
<-modified>
<-includeEmptyFolders>
<-subprop>
<-log log-path | log-path-and-filename | . >
<-disableX11>
<-noexecute>
<-? | --help>
```

Syntax Description for the Batch Export Tool

connection-options
> specifies connection options for the metadata server from which the package is being exported. See "Common Options" on page 313.

> **Requirement**   Required

-package *package-path-and-filename*
> specifies the path and filename of the package file that you want to create. The objects that you export will be placed in this file.

> The filename must have the extension .spk. The path is optional, but if you do specify a path, it must be fully qualified.

> If the path or filename includes spaces, then use quotation marks.

> ### *CAUTION:*
> > **If a file of this name already exists, then it will be overwritten.** If you want to preserve pre-existing packages, then be sure to specify a unique name.

> **Default**   If you do not specify a path, then the package is placed in the directory where the command was launched.

> **Requirement**   You must specify a filename, but the path is optional.

-objects *source-location-1 source-location-2 source-location-3 ... source-location-n*
> specifies the locations of the objects that are to be exported. You can specify any number of locations. Leave a space between each location. If a location includes spaces, then enclose the location in quotation marks.

> Use the following syntax to specify a location:

> */folder-1/folder-2/...folder-n/<object-name(object-type)>*

> The following rules apply to specifying locations:

> * Locations are relative to the **SAS Folders** node. Therefore, the first folder that you specify in a location must be located directly beneath **SAS Folders**.

> * If you specify a folder but you do not specify an object name, then all objects and folders beneath that folder are exported.

> * To export objects from your personal folder, you must specify the actual path (**/User Folders/***user-name***/My Folder** or **/Users/***user-name***/My Folder**) rather than the shortcut (**/My Folder**). Note that the name of the parent folder for user folders varies depending on the SAS release in which the folders were created.

> * If you specify an object name, then you must also specify the object's type. For a list of object types, see "Specifying Object Types" on page 312. Object types are case-sensitive.

Here are some examples of locations:

- -objects/

  This example exports the entire SAS Folders hierarchy and all of its objects.

- -objects*"/User Folders/sasdemo/My Folder"* or -objects*"/Users/sasdemo/My Folder"*

  This example exports all objects and folders that are in the personal folder of the user named sasdemo.

- -objects*"/Shared Data/Orion Star Data/Customer Orders(InformationMap)"*.

  This exports the Customer Orders information map, which is located in **/Shared Data/Orion Star Data**

- -objects*"/Shared Data/Orion Star Data/Customer Orders(InformationMap)" "/Shared Data/Orion Star Data/CUSTOMER_DIM(Table)" "/Shared Data/Orion Star Data/ORDER_FACT(Table)" "/Shared Reports/Orion Star Reports/Sales by Customer Type(Report)"*

  This example exports the Customer Orders information map, the CUSTOMER_DIM table, and the ORDER_FACT table, all of which are located in **/Shared Data/Orion Star Data**; and the Sales by Customer Type report, which is located in **/Shared Reports/Orion Star Reports**.

Requirement	Required
Tips	To further limit the objects that are exported, you can use the -types, -name, -since, and -before options.
	If you want to automatically export all of the objects on which the exported objects depend (instead of having to explicitly list those objects), specify -includeDep.

-types *object-type-1, object-type-2....object-type-n*

specifies a comma-separated list of the types of objects that are to be exported. If you specify this option, then the tool exports only objects that are of the specified types; all other objects are ignored.

If the string contains spaces, then use quotation marks.

For a list of object types, see "Specifying Object Types" on page 312. Object types are case-sensitive.

By default, if you specify an object type that has one or more subtypes, then the subtypes are included in the export. For example, if you specify **Report**, then **Report.Component**, **Report.Image**, and **Report.StoredProcess** are automatically included.

Requirement	Not required.
Interaction	If you want to exclude subtypes of these object types from the export, then use the -excludeSubTypes option with this option.
Note	**Folder** is not a valid value for the -types option. When an object of the specified type is exported, the folder structure that contains it is also exported.

-excludeSubTypes
> specifies that subtypes of the specified object types are to be excluded from the export. In the list of object types, include one or more object types that are specified in the -types option.
>
> For example, suppose you specify the following:
>
> ```
> -types Report,Job -excludeSubtypes
> ```
>
> Objects with the type **Report** and **Job** are included in the export. Objects with the type **Job.Cube**, **Report.Component**, **Report.Image**, and **Report.StoredProcess** are excluded.

Default	If you do not specify -excludeSubTypes with the -types option, then the export automatically includes all subtypes.
Requirement	Not required.

-includeDep
> specifies that all objects on which the exported objects depend are to be included in the export. If you select this option, the export tool recursively identifies dependencies and includes the identified objects in the export.
>
> For example, if you export an information map, the export tool identifies and exports any libraries, tables, and stored processes that are used by the information map. The process also exports any objects that the identified stored processes depend on.
>
> *CAUTION:*
> > **If you do not export the objects on which exported objects depend, then the exported objects might not function properly when you import them to the target metadata server.**

Requirement	Not required
Interaction	The identified dependent objects will include all types of objects, regardless of the values that are specified in the -types option.

-name *character-string*
> specifies a character string that must be present in an object's name in order for the object to be exported. Objects whose names contain the specified string are exported, and all other objects are ignored.
>
> This option only identifies objects. It does not apply to folders.
>
> This option is not case sensitive. For example, if you specify **-name product**, then the tool exports objects with the names **Sales by Product** and **PRODUCT_DIM**.
>
> If the string contains spaces, then use quotation marks.

Requirement	Not required
Interactions	If you specify -includeDesc along with –name, then the object is exported if the text string is present in either the object's name or its description.
	You can use the -nameMatchType option to further define how the -name option is to be applied.

-nameMatchType CONTAINS | EQUALS | STARTSWITH
> specifies how the -name option is to be applied. The following values are valid:

CONTAINS	specifies that the object's name must contain the character string that is specified in -name in order for the object to be exported.
EQUALS	specifies that object's name must exactly match the character string that is specified in -name in order for the object to be exported.
STARTSWITH	specifies that the object's name must begin with the character string that is specified in -name in order for the object to be exported.

The matching of character strings is not case sensitive.

Default	CONTAINS
Requirement	This option is not required.
Interactions	This option is ignored if -name is not specified.
	This option applies only to the object name. If you specify -includeDesc, this option does not apply to the description.

-includeDesc
specifies that the character string that is specified in -name will be searched for in both the name and the description of each object.

Requirement	Not required
Interactions	This option is ignored if -name is not specified.
	The -nameMatchType option does not apply to the object description. It applies only to the object name. For the object description, the CONTAINS match type is always used.

-keywords *keyword*
specifies one or more keywords to be used as filtering criteria. If an object has a keyword that exactly matches one or more of the specified keywords, then the object will be included in the export. If a keyword contains spaces, then put it in quotation marks.

-notes *character-string*
specifies one or more character strings to be used as filtering criteria for public and private Notes. If an object's public or private Notes field contains one or more of the specified strings, then the object will be included in the export. If a string contains spaces, then use quotation marks.

Only certain types of objects (for example, objects used by SAS Data Integration Studio) support the Notes field.

Requirement	Not required

-extName *extended-attribute-name*
specifies the name of an extended attribute. Objects that have an extended attribute that exactly matches the specified attribute name will be included in the export.

Only certain types of objects (for example, objects used by SAS Data Integration Studio) support extended attributes.

Requirement	Not required.

Interaction	You can use -extValue to specify a particular value for the attribute.

-extValue *extended-attribute-value*
specifies a value for the extended attribute that is specified in -extName. Objects that have a value for this attribute that exactly matches the specified value will be included in the export.

Requirement	Not required
Interaction	This option is ignored if -extName is not specified.

-respName *name-of-responsible-user*
specifies a user name that must be present in an object's Responsibilities metadata in order for the object to be exported. The specified name must exactly match the user's Name field in the metadata.

Requirement	Not required.
Interaction	You can use -respRole to specify whether the user is to be an owner or an administrator for the object.

-respRole OWNER | ADMINISTRATOR
specifies the role of the user that is specified in -respName. The valid values are OWNER and ADMINISTRATOR. Objects will be exported only if the object's Responsibilities metadata specifies this role for the user that you specify in -respName.

Requirement	Not required
Interaction	This option is ignored if -respName is not specified.

-since *from-absolute-date | from-relative-date | relative-time-period*
specifies that objects are to be exported only if they were created or modified on or after the specified date, or during the specified time period. You can specify an absolute date, a date relative to the current date, or a time period relative to the current date.

Requirement	Not required
Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
See	"Specifying Dates, Times, and Time Periods" on page 311

-before *from-absolute-date | from-relative-date*
specifies that objects are to be exported only if they were created or modified on or before the specified date. You can specify an absolute date or a date relative to the current date.

Requirement	No
Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
Tip	To specify a date range, specify -since along with this option.
See	"Specifying Dates, Times, and Time Periods" on page 311

-created
   specifies that the -since and -before options are to be based on each object's Created date. If -created is specified, then objects are exported only if the Created date meets the criteria specified in -since and -before.

Default	If -since or -before is specified but -created is not specified, then -modified is assumed.
Requirement	Not required.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-modified
   specifies that the -since and -before options are to be based on each object's Modified date. If -modified is specified, then objects are exported only if the Modified date meets the criteria specified in -since and -before.

   Note: If an object has never been modified, then the object's Modified date is equal to its Created date.

Default	If -since or -before is specified, and if you specify neither -created nor -modified, then -modified is assumed.
Requirement	Not required.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-includeEmptyFolders
   specifies that folders are to be exported even if they do not contain any subfolders or objects.

Default	If this option is not specified, then empty folders are not exported.
Requirement	Not required.

-subprop
   specifies that the export tool is to create an external substitution properties file. By default, the export tool always creates a substitution properties file inside the package file. If you specify this option, then the substitution properties file is created externally.

   In this file, you can modify values that are associated with the exported objects so that the objects will function properly in the target system. Allowable changes include updating properties such as physical directory paths, updating object connections (for example, connections to servers or libraries), and changing an object's location in the SAS Folders tree. For detailed instructions, see "Using the Substitution Properties File" on page 339.

   Note: Instead of specifying connections in the substitution properties file, you can choose to promote the connected objects. For example, when you promote a table, you can also promote the connected library.

Default	If you do not specify this option, then an external substitution properties file is not created. However, a substitution file is still created inside the package file and can be extracted. See "Using the

Requirement	Not required.

-disableX11

enables the batch export tool to execute on UNIX machines when the DISPLAY variable has not been set. This option is new with SAS 9.4M3

Default	This option is disabled by default. If you do not specify this option, then make sure that the DISPLAY variable has been set when running the batch export command on a UNIX machine. See "Set the DISPLAY Variable (UNIX Only)" on page 326.

Requirement	Not required.

-noexecute

specifies that the package file is not to be created. Instead, a list of all of the objects that are included in the export is displayed, and a log file is created. This option is useful if you want to verify that the command is constructed correctly before you execute it.

Example: Batch Export Command

```
ExportPackage -profile "My Profile" -package "c:\SAS Promotion
Files\Package_1.spk" -objects "/Shared Reports/Orion Star
Reports" -includeDep -since "Week to date" -modified -subprop
```

This command does the following:

- exports folders, subfolders, and objects from **/Shared Reports/Orion Star Reports**. Objects are included only if they were created or modified since Monday of the current week.

- recursively identifies the objects on which the exported objects depend, and exports those objects as well.

- exports physical content (for example, SRX files) that is associated with the exported objects.

  Note: When you use the batch tools, the promotion automatically includes all associated content except physical files for tables and external files. (The batch tools do not provide the option of promoting physical files for tables and external files.)

- creates a package file called Package_1.spk that contains the exported objects and content, and places the file in **C:\SAS Promotion Files**.

- creates a substitution properties file called Package_1.subprop, and places the file in **C:\SAS Promotion Files**.

Using the Batch Import Tool

Syntax for the Batch Import Tool

ImportPackage

> *connection-options*
> -package *package-path-and-filename*
> -target *target-metadata-location*
> <-types *object-type-1, object-type-2....object-type-n*>
> <-newOnly>
> <-includeACL>
> <-subprop <*substitution-properties-filename*>>
> <-preservePaths>
> <-disableX11>
> <-noexecute>
> <-expandSummary>
> <-? | --help>

Batch Import Tool Syntax Description

connection-options
> specifies connection options for the metadata server to which the package is being imported. See "Common Options" on page 313.
>
> **Requirement** Required

-package *package-path-and-filename*
> specifies the path and filename of the package file that you want to import. The filename must have the extension .spk. If the path or filename includes spaces, then use quotation marks. The path is optional, but if you do specify a path, it must be fully qualified.
>
> You must specify a package that was created with one of the following methods:
>
> • the batch export tool
>
> • the Export SAS Package wizard (see "Using the Batch Import Tool to Import a Package That Was Created by the Export SAS Package Wizard" on page 344 before you import this type of package)
>
> **Default** If you do not specify a path, then the package is assumed to be located in the directory where the command was launched.
>
> **Requirement** You must specify a filename, but the path is optional.

-target *target-metadata-location*
> the location, relative to **SAS Folders**, to which the package is to be imported on the target metadata server. For example:
>
> -target/   imports the package to the root level (directly under **SAS Folders**).

-target*"/Shared Data"*	imports the package to the **Shared Data** folder.

Note: To import objects to your personal folder, you must specify the actual path (**/User Folders/***user-name***/My Folder** or **/Users/***user-name***/My Folder**) rather than the shortcut (**/My Folder**).

Requirement	Required

-types *object-type-1, object-type-2....object-type-n*

specifies a comma-separated list of the types of objects that are to be imported. If you specify this option, then the tool imports only objects that are of the specified types; all other objects are ignored.

For a list of object types, see "Specifying Object Types" on page 312. Object types are case sensitive.

Requirement	Not required.

Note	**Folder** is not a valid value for the -types option. When an object of the specified type is imported, the folder structure that contains it is also imported.

CAUTION	**You should use the -types option only when you are sure that you want to import only objects of the specified type. Using this option might exclude objects that other objects depend on.** For example, suppose the package includes a table and its associated library. If you specify **Table** in the -types option, the library object will not be imported. For another example, suppose the package includes a table and its associated columns. When using the -types option to import tables with their associated columns, you need to specify the column and table in the -types option.

-newOnly

specifies that objects are to be imported only if they do not already exist in the specified location on the target metadata server. If objects of the same name and type already exist, then they are not imported.

For example, suppose the package file contains a relational information map called **Customer Orders** in the location **/Shared Data/Orion Star Data**, and that the target metadata server contains a relational information map of the same name in that location. If -newOnly is specified, then the information map is not imported.

Default	If you do not specify this option, then objects that already exist on the target metadata server are overwritten by any imported objects that have the same location, name, and type.

Requirement	Not required.

-includeACL

specifies that access controls are to be imported for all of the objects in the package. If the objects already exist on the target metadata server, then any existing access controls for these objects (including access controls that are inherited) will be overwritten by the imported access controls.

Requirement	Not required.

See   "Promoting Access Controls" on page 292.

-subprop *<substitution-properties-filename>*

specifies that imported objects are to be updated based on properties that are specified in the substitution properties file. In this file, you can modify values that are associated with the exported objects so that the objects will function properly in the target system. Allowable changes include updating properties such as physical directory paths, updating object connections (for example, connections to servers or libraries), and changing an object's location in the SAS Folders tree. For detailed instructions, see "Using the Substitution Properties File" on page 339.

Note: Instead of specifying connections in the substitution properties file, you can choose to promote the connected objects. For example, when you promote a table, you can also promote the connected library.

The *substitution-properties-filename* argument is not required with this option. If the filename is omitted, then the name is assumed to be *package-name*.subprop. For example, if your package is named Package1.spk, then the import tool expects the substitution properties file to be named Package1.subprop. If your properties file has a different name, then you must either rename it or specify the filename in the -subprop option. The filename must have the extension .subprop.

CAUTION:

If you do not update the properties file or if you do not specify this option, then some imported objects might not function properly. You might need to log on to the target metadata server after the import is complete and modify servers, paths, and other values that are associated with the imported objects.

Default   If you do not specify this option, the import process will use the source environment's settings for resources such as directory paths, servers, and libraries and will expect those resources to exist in the target environment.

Requirement   Not required.

-preservePaths

specifies that objects and folders are to be placed in the same path where they were located in the source environment. If the folders that are included in the path do not exist, they are created.

For example, suppose you export a library from **/Shared Data/Libraries**. If you import the library to the root level (**SAS Folders**) in the target environment, then the import wizard determines whether a **Libraries** folder exists under **Shared Data**. If the folder does not exist, the wizard creates it and places the imported library in that folder.

If you import objects into a folder below the root level, then this option is ignored.

Note: If you specify both the -noexecute and -expandsummary options when the -preservePaths flag is set, the initial folder structure might be created, but the objects contained within those folders will not be imported.

Requirement   This option is required if you import individual objects (independently of the folders that contain them) and you have specified the root folder (*SAS Folders*) as the import target by specifying **-target /**. This option is necessary because the root folder can contain folders, but it cannot contain individual objects.

-disableX11

enables the batch import tool to execute on UNIX machines when the DISPLAY variable has not been set. This option is new with SAS 9.4M3.

Default	This option is disabled by default. If you do not specify this option, then make sure that the DISPLAY variable has been set when running the batch export command on a UNIX machine. See "Set the DISPLAY Variable (UNIX Only)" on page 326.
Requirement	Not required.

-noexecute

specifies that the package is not to be imported to the target metadata server. Instead, a list of all of the objects that are included in the import is displayed, and a log file is created. This option is useful if you want to verify that the command is constructed correctly before you execute it.

Default	If you do not specify this option, then the package is imported to the target metadata server.
Requirement	Not required

-expandSummary

maps the metadata objects to be imported from the source machine to the target machine, and generates warnings in a log file if the metadata object cannot be mapped

Default	This option is disabled by default.
Requirement	This option is not required.

Using the Substitution Properties File

About the Substitution Properties File

You can use the substitution properties file to modify values that are associated with exported objects so that the objects will function properly after they are imported to the target system. The following types of changes are supported:

- Update an object's connections to other objects. The new objects that you specify must already exist on the target system. For example, you can do the following:

  - connect a table to a different library.

  - connect a stored process, library, OLAP schema, or job to a different application server.

  - connect a stored process to a different source code repository.

  Note: Instead of specifying connections in the substitution properties file, you can choose to promote the connected objects. For example, when you promote a table, you can also promote the connected library.

- update an object's properties. For example, changes to the following properties are supported:

Object Type	Modifiable Properties
Channel or library	Directory path
External file, job, or library	File path
Server	Host, port, or SAS command

Note: Depending on which applications you have installed, you might be able to update additional properties.

Note: You cannot change an object's name.

- change an object's location to a different location in the SAS Folders tree. The location must already exist on the target system.

Create, Edit, and Apply the Substitution Properties File

Follow these steps to create, edit, and apply a substitution properties file:

1. To create an external substitution properties file for objects that you export, specify -subprop in the batch export command.

   Note: If you do not specify -subprop, a substitution file is still created inside the package file and can be extracted. See "Using the Batch Import Tool to Import a Package That Was Created by the Export SAS Package Wizard" on page 344.

   The substitution properties file is created using the same path and filename that you specify in -package, except that it has the file extension .subprop instead of .spk. For example, if you specify the package name Package_1.spk, then the substitution properties file is named Package_1.subprop.

2. Open the substitution properties file in a text editor, and make any necessary updates to connections, properties, or locations for the exported objects. For detailed instructions, see the following sections:

   - "Use Substitution Properties to Update Metadata Connections" on page 340

   - "Use Substitution Properties to Update an Object's Properties" on page 341

   - "Use Substitution Properties to Specify New Folder Locations for Objects" on page 342

3. To apply the changes that you made to the substitution properties file, specify the -subprop option when you run the batch import tool.

Use Substitution Properties to Update Metadata Connections

Follow these steps to use the substitution properties file to modify an object's connections to other objects:

1. In the substitution properties file, locate the sections that contain connection information. Here are examples of section titles for connections:

   ```
   [Connections: Application Server]
   ApplicationServer[1].SourceName=SASAppTest
   ApplicationServer[1].TargetName=SASAppTest

   [Connections: Library]
   Library[1].SourcePath=/Shared Data/Oracle Tables/myoratest(Library)
   ```

```
Library[1].TargetPath=/Shared Data/Oracle Tables/myoratest(Library)

[Connections: Source Code Repository]
SourceCodeRepository[1].ApplicationServer=ApplicationServer[1]
SourceCodeRepository[1].SourceDirectory=c:\\StoredProcessesTest
SourceCodeRepository[1].TargetDirectory=c:\\StoredProcessesTest
SourceCodeRepository[1].CreateIfNeeded=false
```

2. Update the target connections as needed. Specify the exact names of resources as they appear on the target system. The names are case sensitive. The objects that you specify (except source code repositories) must already exist on the target system.

   Modify only entries that have the word **Target** in the label (for example, **ApplicationServer[1].TargetName=object-name**). Do not modify source connections. For example, do not modify **ApplicationServer[1].SourceName=object-name**.

   Here are examples of modified connections to an application server, a library, and a source code repository:

```
Connections: Application Server]
ApplicationServer[1].SourceName=SASAppTest
ApplicationServer[1].TargetName=SASApp

[Connections: Library]
Library[1].SourcePath=/Shared Data/Oracle Tables/myoratest(Library)
Library[1].TargetPath=/Shared Data/Oracle Tables/myora(Library)

[Connections: Source Code Repository]
SourceCodeRepository[1].ApplicationServer=ApplicationServer[1]
SourceCodeRepository[1].SourceDirectory=c:\\StoredProcessesTest
SourceCodeRepository[1].TargetDirectory=c:\\StoredProcesses
SourceCodeRepository[1].CreateIfNeeded=true
```

   Note that **CreateIfNeeded** has been changed to **true**. This means that the metadata definition for the source code repository will be created if it does not exist on the target system. (The physical location for the source code must already exist.)

   Note: The **CreateIfNeeded** property is used only for stored process objects.

Use Substitution Properties to Update an Object's Properties

Follow these steps to update an object's properties in the substitution properties file:

1. In the substitution properties file, locate the section for the object whose properties you want to change. Here are examples of section headings for a library and an application server:

```
[/Shared Data/mylib2/mylib2(Library)]
DirPath=/tmp/mylib2

[/System/Servers/SASApp2(ApplicationServer)]

Workspace Server.Command=/usr/local/SAS/config/Lev1/SASApp/WorkspaceServer/WorkspaceServer.sh

# The following SAS Application Server properties are provided for
# context only and may not be modified.

Workspace Server.Command.Server=SASApp2 - Workspace Server
```

```
Workspace Server.Port[1]=8591


# The following SAS Application Server properties are provided for
# context only and may not be modified.


Workspace Server.Port[1].Server=SASApp2 - Workspace Server


Workspace Server.Host[1]=host1.example.com
```

2. Within each object's section, update properties to values that are valid in the target environment. The resources that you specify (for example, directory paths) must already exist on the target system. Here are examples of a modified directory path for a library and a modified port and host name for a server:

```
[/Shared Data/MyLib/mylib(Library)]
DirPath=/Data/mylib2


[/System/Servers/SASApp2(ApplicationServer)]


Workspace Server.Command=/usr/local/SAS/config/Lev2/SASApp/WorkspaceServer/WorkspaceServer.sh


# The following SAS Application Server properties are provided for
# context only and may not be modified.


Workspace Server.Command.Server=SASApp2 - Workspace Server


Workspace Server.Port[1]=8592


# The following SAS Application Server properties are provided for
# context only and may not be modified.


Workspace Server.Port[1].Server=SASApp2 - Workspace Server


Workspace Server.Host[1]=host2.example.com
```

Note: The message `The following SAS Application Server properties are provided for context only and may not be modified` applies only to the line that immediately follows the message. Any properties that are not preceded by this message can be updated.

Use Substitution Properties to Specify New Folder Locations for Objects

The batch import tool enables you to reorganize content as part of the promotion process. You can do so by specifying new folder locations for individual objects that you are importing, as follows:

1. Determine whether the substitution properties file contains a section for the object that you want to move. For example, the following section title applies to the Orion Star library that is located in **Shared Data/Orion Star Data**:

```
[/Shared Data/Orion Star Data/Orion Star(Library)]
```

If you do not find a section title for the object that you want to move, then add a section title to the file using the following syntax. Be sure to include the entire path, relative to **SASFolders**. (You can obtain the path information from the export log, if necessary.)

```
[path-relative-to-SASFolders/object-name(object-type)]
```

2. Immediately following the object's section title, add an entry that specifies the folder to which you want to move the object. The specified folder must already exist on the target metadata server. Be sure to include the entire path, relative to **SASFolders**. Use the following syntax:

```
Framework.FolderPath=path-relative-to-SASFolders(Folder)
```

Add these entries for each object or folder that you want to move. For example:

- Suppose **/Shared Data/Orion Star Data** on the source metadata server contains three tables (CUSTOMER_DIM, ORDER_FACT, and PRODUCT_DIM) and two information maps (Customer Orders and Product Sales). To move these objects to an already-existing folder called **/Shared Data/Imported Data** on the target metadata server, you would update the substitution properties file with the following entries:

```
[/Shared Data/Orion Star Data/CUSTOMER_DIM(Table)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)

[/Shared Data/Orion Star Data/ORDER_FACT(Table)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)

[/Shared Data/Orion Star Data/PRODUCT_DIM(Table)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)

[/Shared Data/Orion Star Data/Customer Orders(InformationMap)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)

[/Shared Data/Orion Star Data/Product Sales(InformationMap)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)
```

On the target system, the three tables will appear directly under the **Imported Data** folder.

- If you want to move the entire **/Shared Data/Orion Star Data** folder and its contents to an already-existing folder called **/Shared Data/Imported Data** on the target metadata server, you would update the substitution properties file with the following entries:

```
[/Shared Data/Orion Star Data(Folder)]
Framework.FolderPath=/Shared Data/Imported Data(Folder)
```

The imported objects will appear in the path **/Shared Data/Imported Data/Orion Star Data** on the target system.

Note: If you used the -includeDep option to export an object's dependent objects, the dependent objects are not moved automatically. You must move the dependent objects separately.

Example: Batch Import Command

```
ImportPackage -profile "My Profile" -package "c:\SAS Promotion
Files\Package_1.spk" -target / -include ACL-subprop
```

This command does the following:

- imports folders, subfolders, and objects from **Package_1.spk** and writes them to the root location under **SAS Folders** on the target metadata server

- updates objects on the target metadata server with values that are specified in the substitution properties file

- imports access controls for all of the objects in the package

- imports physical content (for example, SRX files) that is associated with the imported objects

  Note: When you use the batch tools, the promotion automatically includes all associated content except physical files for tables and external files. (The batch tools do not provide the option of promoting physical files for tables and external files.)

Viewing the Batch Export or Import Log

When you run the batch export or import tool, a log file is created in the following path:

- On Windows Vista and Windows 7 systems: **C:\Users***user***\AppData\Roaming\SAS\Logs**

- On other Windows systems: **C:\Documents and Settings***user***\Application Data\SAS\Logs**

- On UNIX systems: *user's-home-directory***/SAS/Logs**

Note:

- To see the log file on Windows systems, you must enable the option to view hidden files and folders.

- If you want to direct the log to a different location, specify the -log option with the import or export command.

The log file is named **Export_***yyMMddHHmmss***.log** or **Import_***yyMMddHHmmss***.log**, where *yyMMddHHmmss* is the date and time that the command was executed.

If the batch import or export command fails, you can use the log file to determine the source of the error. For a list of common error messages, see "Troubleshooting the Promotion Tools" on page 305.

Using the Batch Import Tool to Import a Package That Was Created by the Export SAS Package Wizard

You can use the batch import tool to import a package that was created by the export wizard. When you run the export wizard, it creates a substitution properties file and places it in the package. If you want to provide different values for servers, paths, or other resources, then you must extract the substitution file and modify the properties. Follow these steps:

1. Extract the substitution properties file from the package.

   CAUTION:

> **Do not save the package file after you perform the extract. Packages cannot be imported after they have been saved using a process other than the promotion tools.**

2. Modify the target values in the substitution properties file as needed. See "Using the Substitution Properties File" on page 339.

3. (Optional) Rename the properties file so that it has the same base filename as the package. For example, if your package is named Package1.spk, then rename the properties file Package1.subprop.

4. Specify -subprop in the batch import command. If you did not rename the package (as described in the preceding step), then you must specify the substitution properties filename with this option.

Using the Import SAS Package Wizard to Import a Package That Was Created by the Batch Export Tool

You can use the import wizard to import a package that was created by the batch export tool. The import works the same as it would if the package had been created by the export wizard.

Chapter 26
Using the Batch Relationship Reporting Tools

About the Batch Relationship Reporting Tools

The batch relationship reporting tools were introduced in SAS 9.4M1. These tools identify relationships among the content objects in the SAS Folders tree. For example, you can identify the objects that a given object depends on or contains; the objects that depend on or contain a given object; and the objects that are associated with a given object. Both direct and nested relationships can be identified. Use of the batch relationship reporting tools involves the following steps:

1. First, you use the Relationship Loader batch tool (sas-relationship-loader) to scan objects, retrieve their relationship information, and load the information into a database. This tool is located in **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin**. For detailed usage and syntax information, see "Using the Relationship Loader (sas-relationship-loader)" on page 349.

   Effective with SAS 9.4M3, relationship data loading is scheduled by default to execute on an hourly basis. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.

   Note: Effective with SAS 9.4M3, you can also load third-party lineage information to the relationship database. For details, see "Using the SAS Metadata Bridge Relationship Loader (sas-metabridge-relationship-loader)" on page 368.

2. Then you use the Relationship Reporter batch tool (sas-relationship-reporter) to produce a relationship report for selected objects. You can choose from four standard reports: lineage, impact, direct dependencies, and indirect dependencies. Custom options enable you to produce more fine-tuned results. This tool is located in **SAS-installation-directory/SASPlatformObjectFramework/9.4/tools**. For detailed usage and syntax information, see "Using the Relationship Reporter (sas-relationship-reporter)" on page 353.

About SAS Lineage

SAS Lineage is a web-based diagram component for visualizing relationships between objects. This feature, which became available in October 2014, is used by SAS software products including SAS DataFlux Data Management Studio, SAS Decision Manager, SAS Model Manager, and SAS Business Rules Manager. The component has two modes. First, a network diagram displays all relationships. Second, two left-to-right dependency diagrams are available: one that displays data governance information (governance) and another that displays parent and child relationships only (impact analysis).

The relationship information displayed in these diagrams is drawn from the same relationship database that is a used by the batch relationship reporting tools.

For more information, see SAS Lineage: User's Guide.

Using the Relationship Loader (sas-relationship-loader)

About the Relationship Loader

The Relationship Loader batch tool (sas-relationship-loader) enables you to scan content objects in the SAS Folders tree, retrieve the objects' relationship information, and store the information in a database on the middle tier so that it is available for reporting. You can specify folder locations or filtering options to limit the objects that are scanned. In addition, you can specify the -clean option to remove orphaned objects from the database.

The command **sas-relationship-loader** is located in the following path:

SAS-installation-directory/SASPlatformObjectFramework/9.4/ tools/admin

You can execute this command on any machine where the SAS Platform Object Framework is installed.

Note: Effective with SAS 9.4M3, relationship data loading is scheduled by default to execute on an hourly basis. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.

Syntax for the Relationship Loader

sas-relationship-loader

> *web-connection-options*
> > *object-paths* | *content-filters* | -loadAll | -clean
> > <-log *log-path* | *log-path-and-filename*>

T I P   All command options are case sensitive.

Syntax Description for the Relationship Loader

web-connection-options
specifies connection options to log on to the SAS Web Server or SAS Web Application Server.

Restriction	For this command, you must specify an unrestricted user (for example, sasadm@saspw).
Requirement	Required
See	"Specifying Web Connection Options" on page 363

-log *log-path* | *log-path-and-filename*
specifies the path (or the path and filename) where the log file is to be written. If the path or filename contains spaces, then enclose the value in double quotation marks.

Requirement	Not required.

object-paths

specifies the path locations, relative to **SAS Folders**, of the objects whose relationship data is to be loaded. You can specify any number of locations. Leave a space between each location. If a location includes spaces, then enclose the location in double quotation marks.

Use the following syntax to specify each location:

/folder-1/folder-2/...folder-n/<object-name(object-type)>

Restriction	You cannot specify this option with *content-filters*, -loadAll, or -clean.
Requirement	You must specify one of the following: *object-paths*, *content-filters*, -loadAll, or -clean.
See	For additional details and examples, see "Specifying Object Paths" on page 363.

content-filters

specifies filters to select the objects whose relationship data is to be loaded. Relationship data is loaded only for objects that meet the specified criteria. You can use the following arguments to specify a filter:

```
<-before to-absolute-date | to-relative-date>
<-created>
<-excludeSubTypes>
<-folder folder-path>
<-folderTree folder-path>
<-includeDesc>
<-keywords keyword>
<-modified>
<-name character-string>
<-nameMatchType CONTAINS | EQUALS | STARTSWITH>
<-respName name-of-responsible-user>
<-respRole OWNER | ADMINISTRATOR>
<-since from-absolute-date | from-relative-date|relative-time-period>
<-types object-type-1, object-type-2....object-type-n>
```

Restriction	You cannot specify this option with *object-paths*, -clean, or -loadAll.
Requirement	You must specify one of the following: *object-paths*, *content-filters*, -loadAll, or -clean.
See	For a description of each argument, see "Specifying Filtering Arguments" on page 364.

-loadAll

specifies that relationships are to be loaded for all content objects in the SAS Folders tree. This option can require significant processing time.

TIP All command options are case sensitive.

Note Effective with SAS 9.4M3, relationship data loading is scheduled by default to execute on an hourly basis. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.

Restriction	You cannot specify this option with *content-filters*, *object-paths*, or -clean.
Requirement	You must specify one of the following: *object-paths*, *content-filters*, -loadAll, or -clean.

-clean

specifies that orphaned resources are to be removed from the relationship database. For example, if a user has deleted a table object from the metadata repository, the -clean option removes that object and its relationship information from the relationship database.

Note	Effective with SAS 9.4M3, cleaning of relationship data is scheduled by default to execute daily at 11:00 p.m. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.
Restriction	You cannot specify this option with *content-filters*, *object-paths*, or -loadAll.
Requirement	You must specify one of the following: *object-paths*, *content-filters*, -loadAll, or -clean.

Best Practices for Using the Relationship Loader

The first time you run the Relationship Loader tool, consider specifying the -loadAll option so that relationships are loaded for all content objects in the SAS Folders tree. Doing so ensures that the Relationship Reporter tool (sas-relationship-reporter) has all of the information it needs to produce accurate and complete reports.

If you want to limit processing time in subsequent executions of the Relationship Loader, you can use filtering criteria to scan only those objects that were created or modified since the date of the previous scan. Occasional use of the -clean option helps control the size of the relationship database.

You can simplify administration by configuring regularly scheduled loading and cleaning of relationship data. Effective with SAS 9.4M3, relationship loading and cleaning is scheduled to execute regularly by default. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.

Examples: sas-relationship-loader

The following command loads relationship data for only the Hello World sample stored process:

```
sas-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
"/Products/SAS Intelligence Platform/Samples/Sample: Hello World(StoredProcess)"
```

Note: 7980 is the default port number for the SAS Web Application Server. The port number might be different in your deployment.

The following command loads relationship data for all tables that are contained within the **/Shared Data** folder:

```
sas-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
-types Table -folderTree "/Shared Data"
```

The following command loads relationship data for all content objects:

```
sas-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
-loadAll
```

Configuring Automatic Relationship Loading and Cleaning

Effective with SAS 9.4M3, automatic loading of relationship data is configured by default to execute on an hourly basis. The load process scans the SAS Folders tree for content items that were created or modified since the last scheduled load operation. The process retrieves these objects' relationship information and stores it in the relationship database.

Effective with SAS 9.4M3, cleaning of relationship data is configured by default to execute daily at 11:00 p.m. The cleaning operation removes relationship information for objects that have been deleted from your metadata repository.

To configure a different schedule (or to set the schedule if you are using a release earlier than SAS 9.4M3), follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, select **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Web Infra Platform Services 9.4** .

2. Under **Web Infra Platform Services 9.4**, right-click **RelationshipContentService** and select **Properties**.

3. In the RelationshipContentService Properties dialog box, select the **Settings** tab. In the left panel of the **Settings** tab, select **Scheduling**.

4. Follow these steps to schedule the loading process:

    a. Under **Scheduling for Load Task Enabled**, select **true** in the drop-down box.

    b. Select a **Start** time only if you want to delay the initial execution for some reason.

        Note: Since the delayed **Start** time takes effect every time the SAS Web Application Server is restarted, you might want to remove this selection after the initial execution.

        If you do not specify a **Start** time, the initial execution will occur immediately after you restart the SAS Web Application Server.

    c. Under **Recurrence,** select **Hourly**, **Daily**, **Weekly**, or **Custom**. If you select **Custom**, enter the scheduling interval as a number of minutes under **Frequency**.

5. To schedule the cleaning process, select **true** in the drop-down box under **Scheduling for Clean Task Enabled**. Then respond to the prompts, which are similar to those for the load task.

6. Click **OK**.

7. To put the schedule into effect, restart the SAS Web Application Server.

Using the Relationship Reporter (sas-relationship-reporter)

About the Relationship Reporter

Use the Relationship Reporter batch tool (sas-relationship-reporter) to create reports about the relationships among content objects in the SAS Folders tree. The following features are provided:

- You can choose from four standard reports: lineage, impact, direct dependencies, and indirect dependencies. Custom options enable you to produce more fine-tuned results.

- To select the objects whose relationships you want to report (referred to as *subjects*), you can specify one or more locations in the SAS Folders tree, or you can specify filtering criteria. You can also specify filters to limit the report output.

- Reports can be written to the console or to a file. You can specify a comma-separated values (CSV) or text file format.

The command **sas-relationship-reporter** is located in the following path:

***SAS-installation-directory*/SASPlatformObjectFramework/9.4/ tools**

You can execute this command on any machine where the SAS Platform Object Framework is installed.

Prerequisites for Using the Relationship Reporter

Before you can use the Relationship Reporter batch tool, your content must be analyzed and the resulting relationship data must be loaded to a database. Administrators can use either of the following methods to load this data:

- Run the sas-relationship-loader command. This command includes filters that you can use to limit the amount of content that is analyzed. See "Using the Relationship Loader (sas-relationship-loader)" on page 349.

- Schedule a job to load your relationship data on a regular basis. Effective with SAS 9.4M3, relationship data loading is scheduled by default to execute on an hourly basis. See "Configuring Automatic Relationship Loading and Cleaning" on page 352.

Be sure to load relationship data for all of the objects that might be related to the objects that you plan to report on. See "Best Practices for Using the Relationship Loader" on page 351 and "Scope of Relationship Reports" on page 355.

About the Predefined Relationship Reports

For ease of use, the following predefined reports are provided with the Relationship Reporter batch tool:

lineage report
> reports the objects that each subject depends on or contains. By default, nested dependencies and contents are reported to a depth of 10. For example, if the subject is an information map, the report might list the following:
>
> - the tables that the map depends on. The report would also list the libraries that the tables depend on and the columns that each table contains.
>
> - the information map data items and joins that the map contains. The report would also list the column that each identified data item and join depends on.
>
> *Note:* You might have additional types of dependent or contained objects for these object types.

impact report
> reports objects that depend on or contain each subject. You can use this report to identify the objects that would be affected if the subject were to be changed or deleted. By default, nested dependencies and contents are reported to a depth of 10. For example, if the subject is an information map, then the SAS Web Report Studio reports that depend on the information map would be listed. If the subject is a column, then the report might list the following:
>
> - the tables that contain the column. The report would also list information maps that depend on each identified table, as well as SAS Web Report Studio reports that depend on each identified information map.
>
> - the information map data items and joins that depend on the column. The report would also list information maps that contain the identified data items and joins.

direct dependencies
> reports the objects that each subject directly depends on. By default, nested dependencies are not reported. For example, if the subject is a table, the report would list the library that the table depends on. If the subject is an information map, the report would list the tables and the library that the information map depends on.

indirect dependencies
> reports objects that each subject depends on either directly or indirectly. It lists the subject's lineage to each leaf (outermost) node that has a dependency relationship to its immediate parent. Intermediary relationships are reported even if they are of a different type. By default, the report excludes lineage to leaf nodes that are contained by or associated with their immediate parents.
>
> For example, if the subject is an information map, the report might include the following:
>
> - the tables that the map depends on. The report would also list the library that each identified table depends on. Columns that are contained by the tables would not be reported (because the columns are leaf nodes that do not have a dependency relationship with their immediate parents).
>
> - the library that the map depends on.
>
> - the information map data items and joins that the map contains (because these are intermediate relationships, not leaf nodes). The report would also list the column that each identified data item and join depends on (because these columns are leaf nodes that have a dependency relationship with their immediate parents).

Note: The relationships that are listed in the predefined reports are limited based on the scope of the relationship loading process and based on the connecting user's permissions. For details, see the next topic.

Scope of Relationship Reports

The related objects that are reported are limited based on the scope of the relationship loading process, as follows:

- On lineage, dependency, and indirect dependency reports (and custom reports with a direction of TO), nested relationships are reported only for objects that were included in the scope of the relationship loading process. For example, suppose you have loaded relationship data for Folder A, but not for Folders B and C. You then run a lineage report for an information map in Folder A. The report would list the map's relationships to tables, libraries, and data items in Folder B and Folder C. However, it would not list nested dependencies for those related objects.

- On impact reports (and custom reports with a direction of FROM), related objects are reported only if they were included in the scope of the relationship loading process. For example, suppose you have loaded relationship data for Folder A, but not for Folders B and C. You then run an impact report for a table in Folder A. The report would not list the table's relationships to information maps, cubes, and SAS Web Report Studio reports in Folder B and Folder C.

For tips to ensure that the appropriate objects are included in the loading process, see "Best Practices for Using the Relationship Loader" on page 351.

Report content is also limited based on the permissions of the user who is specified in the connection options. For example, suppose you run a lineage report for an information map and you specify a connecting user who has ReadMetadata permission on the information map but is denied ReadMetadata permission on the map's related tables. The report output would indicate that the map depends on tables. However, it would not specify the tables' names, and it would not list the columns that the tables contain.

Best Practices for Using the Relationship Reporter

When you are starting out with the Relationship Reporter, it is recommended that you use the -report option to specify one of the predefined reports. These reports, which should meet the needs of most users, are described in the preceding topic.

After gaining experience with the predefined reports, you might want to use the relationship filter (-rel) options to obtain more fine-tuned results. You can use these options to override the default settings of the standard reports or to create custom reports. For example, you can specify a relationship direction, a reporting depth, and the types of relationships to be reported. Use these options with caution to avoid performance issues. If necessary, you can also use the filtering criteria available in the -relContent option to limit the report content.

Syntax for the Relationship Reporter

sas-relationship-reporter

  connection-options
    <-report *report-type*>
    <-file *output-path-and-filename*>
    <-format LIST | CSV>
    <-log *log-path* | *log-path-and-filename* >
    <-rel {*relationship-filters*}>
    <-relContent {*relationship-content-filters*}>
    <*object-paths*> | <*content-filters*>

> `TIP` All command options are case sensitive.

Syntax Description for the Relationship Reporter

web-connection-options

specifies connection options to log on to the SAS Web Server or SAS Web Application Server.

Requirement	Required
Note	You can specify any authorized user for this command. However, the report output will be limited based on the user's permissions.
See	"Specifying Web Connection Options" on page 363

-report *report-type*

specifies the type of report to be generated. The following values are valid:

lineage	reports objects that each subject depends on or contains. The default settings for this report are as follows: `-rel {-types D I -depth 10 -direction TO -traversal PRE}`
impact	reports objects that depend on or contain each subject. The default settings for this report are as follows: `-rel {-types D I -depth 10 -direction FROM -traversal PRE}`
directDependencies	returns the objects that each subject directly depends on. The default settings for this report are as follows: `-rel {-types D -depth 1 -direction TO -traversal PRE}`
indirectDependencies	returns the objects that each subject depends on either directly or indirectly. The default settings for this report are as follows: `-rel {-types D –depth 10 –direction TO –traversal POST}`
custom	returns relationship information based on the user's -rel option settings.

For detailed descriptions, see "About the Predefined Relationship Reports" on page 353 and "Scope of Relationship Reports" on page 355.

Default	*custom*
Interaction	You can use the -rel option settings to override a report's default settings.

-file *output-path-and-filename*

specifies the path and name of a file to which the output is to be written.

Default	If no path or filename is specified, the output is written to the console. If you specify a filename but no path, the file is written to the current working directory. If you specify a path but no filename, a file called RelationshipReport_*xxxxxxxxxxxx* is created with the extension .txt or .csv.

-format LIST | CSV
  specifies the format in which the output should be written. The following values are valid:

  LIST writes the output in a text format.

  CSV writes the output in a comma-separated values format.

  Default LIST

-log *log-path* | *log-path-and-filename*
  specifies the path (or the path and filename) where the log file is to be written. If the path or filename contains spaces, then enclose the value in double quotation marks.

-rel {*relationship-filters***}**
  specifies filters that define the relationship types and scope that are to be included in the report output. The report will include relationships only if they meet the specified criteria. You can use the following arguments to specify a relationship filter. The argument list must be enclosed in braces ({ and }).

  <-attrName *resource-attribute-name*>
  <-attrValue *resource-attribute-value*>
  <-depth *number-of-dependency-levels*>
  <-direction TO | FROM>
  <-traversal PRE | POST >
  <-types *relationship-type-1, relationship-type-2....relationship-type-n*>

  -attrName *resource-attribute-name*
    specifies the name of a resource attribute. Objects that have a resource attribute that exactly matches the specified attribute name and value will be included.

    Only certain types of objects (for example, objects used by SAS Data Integration Studio) support resource attributes.

    Interaction You must also use -attrValue to specify a particular value for the attribute.

  -attrValue *resource-attribute-value*
    specifies a value for the resource attribute that is specified in -attrName. Objects that have a resource attribute that exactly matches the specified attribute name and value will be included.

    Interaction This option is ignored if -attrName is not specified.

  -depth *number-of-dependency-levels*
    specifies the number of levels of nested dependencies to return.

    Default For the defaults used in the predefined reports, see "-report *report-type*" on page 356. For custom reports, a depth of 1 is assumed.

  -direction TO | FROM
    specifies the relationship direction to process, relative to the objects (subjects) whose relationships are being reported.

    TO reports related objects that the subject consumes. This direction is used for lineage-type reports, which identify objects that contribute to or affect the subject.

FROM   reports related objects that are consumed by the subject. This
         direction is used for impact-type reports, which identify objects
         that would be affected if the subject were to be changed or deleted.

Note: The related objects that are reported are limited based on the scope of the
         relationship loading process. See "Scope of Relationship Reports" on page
         355.

Default   For the defaults used in the predefined reports, see "-report *report-type*"
          on page 356. For custom reports, a value of *TO* is assumed.

-traversal PRE | POST
   specifies the traversal policy that is used when applying the relationship content
   filtering (-relContent) options. The traversal policy determines the order in which
   the related objects are evaluated against the filtering criteria. The following
   values are valid:

PRE   specifies the use of a pre-order (or "top down") traversal policy to
         filter relationship content. Each subtree is evaluated recursively,
         starting with the left-most child node.

POST   specifies the use of a post-order (or "bottom up") traversal to filter
         relationship content. Each subtree is evaluated recursively, starting
         with the left-most leaf node.

Default   For the defaults used in the predefined reports, see "-report *report-type*"
          on page 356. For custom reports, a value of *PRE* is assumed.

-types *relationship-type-1, relationship-type-2....relationship-type-n*
   specifies the types of relationships that are to be included in the report. The
   following values are valid:

D   specifies the dependency relationship type. If Object 1 either cannot
         function without Object 2 or cannot be defined without Object 2, then
         the two objects have a dependency relationship. For example, tables
         depend on libraries; jobs can depend on tables; and reports can depend
         on cubes, information maps, stored processes, and tables.

I   specifies the contains relationship type. If Object 1 contains Object 2,
         and if Object 2 cannot exist or is unusable without Object 1, then the
         two objects have a contains relationship. For example, tables can
         contain columns; jobs can contain job transformations; and information
         maps can contain data items, filters, and joins.

A   specifies the association relationship type. In this relationship type,
         Object 1 is associated with Object 2 through a known or unknown
         system or method. For examples, libraries, deployed jobs, and OLAP
         schemas can be associated with servers; and stored processes can be
         associated with servers and source code repositories.

P   specifies the parent-child relationship type, in which Object 1 is the
         parent of Object 2. Object 2 is the child of Object 1, and Object 2
         cannot be found in the hierarchy unless Object 1 is present.

E   specifies the equal relationship type, in which Object 1 and Object 2
         represent the same thing but are accessed through different methods or
         systems.

S specifies the synonymous relationship type, in which Object 1 and Object 2 are alike in meaning or significance.

Default Custom reports default to the *D* and *A* relationships types. For the defaults used in the predefined reports, see "-report *report-type*" on page 356.

-relContent {*relationship-content-filters*}
  specifies filters to select the related objects that are to be included in the report output. The report will include related objects only if they meet the specified criteria.

  You can use the following arguments to specify a relationship content filter. The argument list must be enclosed in braces ({ and }).

  <-attrName *resource-attribute-name*>
  <-attrValue *resource-attribute-value*>
  <-before *to-absolute-date | to-relative-date*>
  <-created>
  <-excludeSubTypes>
  <-folder *folder-path*>
  <-folderTree *folder-path*>
  <-includeDesc>
  <-keywords *keyword*>
  <-modified>
  <-name *character-string*>
  <-nameMatchType CONTAINS | EQUALS | STARTSWITH>
  <-respName *name-of-responsible-user*>
  <-respRole OWNER | ADMINISTRATOR>
  <-since *from-absolute-date | from-relative-date|relative-time-period*>
  <-types *object-type-1, object-type-2....object-type-n*>

  See For a description of each argument, see "Specifying Filtering Arguments" on page 364.

---

object-paths
  specifies the path locations, relative to **SAS Folders**, of the objects whose relationships are to be reported.

  You can specify any number of locations. Leave a space between each location. If a location includes spaces, then enclose the location in double quotation marks. Use the following syntax to specify a location:

  /folder-1/folder-2/...folder-n/<object-name(object-type)>

  Restriction You cannot specify this option with *content-filters*.

  See For additional details and examples, see "Specifying Object Paths" on page 363.

---

content-filters
  specifies filters to select objects for which relationships are to be reported. Relationships will be reported only for objects that meet the specified criteria. You can use the following arguments to specify a content filter:

  <-attrName *resource-attribute-name*>
  <-attrValue *resource-attribute-value*>
  <-before *to-absolute-date | to-relative-date*>
  <-created>
  <-excludeSubTypes>
  <-includeDesc>

> <-keywords *keyword*>
> <-modified>
> <-name *character-string*>
> <-nameMatchType CONTAINS | EQUALS | STARTSWITH>
> <-respName *name-of-responsible-user*>
> <-respRole OWNER | ADMINISTRATOR>
> <-since *from-absolute-date | from-relative-date|relative-time-period*>
> <-types *object-type-1, object-type-2....object-type-n*>

Restriction	You cannot specify this option with *object-paths*.
See	For a description of each argument, see "Specifying Filtering Arguments" on page 364.

Example: Lineage Report

The following command produces a lineage report for an information map with the name Product Orders. The report shows all of the objects that the information map either depends on or contains. The report is written in list format to the file ProductOrdersLineage.

```
sas-relationship-reporter -host host01.example.com -port 7980 -user domain1\user123 -password pw123
-report lineage -types InformationMap -name "Product Orders" -file ProductOrdersLineage
```

Note: 7980 is the default port number for the SAS Web Application Server. The port number might be different in your deployment.

Here is the output from this command:

```
"/Shared Data/Orion Star Data/Product Orders" (Information map (relational))
  --> Is dependent on: "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
      --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
      --> Contains: "CostPrice_Per_Unit" (Column)
      --> Contains: "Customer_ID" (Column)
      --> Contains: "Delivery_Date" (Column)
      --> Contains: "Discount" (Column)
      --> Contains: "Order_Date" (Column)
      --> Contains: "Order_ID" (Column)
      --> Contains: "Order_Type" (Column)
      --> Contains: "Product_ID" (Column)
      --> Contains: "Quantity" (Column)
      --> Contains: "Street_ID" (Column)
      --> Contains: "Total_Retail_Price" (Column)
  --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
  --> Is dependent on: "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
      --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
      --> Contains: "Product_Category" (Column)
      --> Contains: "Product_Group" (Column)
      --> Contains: "Product_ID" (Column)
      --> Contains: "Product_Name" (Column)
      --> Contains: "Supplier_ID" (Column)
      --> Contains: "Supplier_Name" (Column)
  --> Contains: "Customer Id" (InformationMapDataItem)
      --> Is dependent on: "Customer_ID" (Column) - in "/Shared Data/Orion Star DataORDER_FACT" (Table)
  --> Contains: "Date Order Was Placed By Customer" (InformationMapDataItem)
      --> Is dependent on: "Order_Date" (Column) - in "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
  --> Contains: "Inner join1" (InformationMapJoin)
```

```
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
   --> Contains: "Product Group" (InformationMapDataItem)
        --> Is dependent on: "Product_Group" (Column) - in "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
   --> Contains: "Product Id" (InformationMapDataItem)
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
   --> Contains: "Product Id2" (InformationMapDataItem)
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
   --> Contains: "Product Name" (InformationMapDataItem)
        --> Is dependent on: "Product_Name" (Column) - in "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
   --> Contains: "Quantity Ordered" (InformationMapDataItem)
        --> Is dependent on: "Quantity" (Column) - in "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
   --> Contains: "Total Retail Price For This Product" (InformationMapDataItem)
        --> Is dependent on: "Total_Retail_Price" (Column) - in "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
```

Note: If your deployment is not current with SAS 9.4M2, your reports will use the following terms to describe the relationships shown in these examples:

- **Depends** (instead of **Is dependent on**)

- **Includes** (instead of **Contains**)

- **Associated** (instead of **Is associated with**)

Example: Lineage Report with Content Filtering

The following command produces a lineage report that is similar to the previous example, except that the output is filtered to include only those related objects that have **product** in the object name. The report is written in list format to the file ProductOrdersLineage_Filtered.

```
sas-relationship-reporter -host host01.example.com -port 7980 -user domain1\user123 -password pw123
-report lineage -types InformationMap -name "Product Orders" -file ProductOrdersLineage_Filtered
-relContent {-name "Product"}
```

Here is the output from this command:

```
"/Shared Data/Orion Star Data/My Folder/Product Orders" (Information map (relational))
  --> Is dependent on: "/Shared Data/Orion Star Data/My Folder/PRODUCT_DIM" (Table)
        --> Contains: "Product_Category" (Column)
        --> Contains: "Product_Group" (Column)
        --> Contains: "Product_ID" (Column)
        --> Contains: "Product_Name" (Column)
  --> Contains: "Product Group" (InformationMapDataItem)
        --> Is dependent on: "Product_Group" (Column) - in "/Shared Data/Orion Star Data/My Folder/PRODUCT_DIM" (Table)
  --> Contains: "Product Id" (InformationMapDataItem)
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/My Folder/ORDER_FACT" (Table)
  --> Contains: "Product Id2" (InformationMapDataItem)
        --> Is dependent on: "Product_ID" (Column) - in "/Shared Data/Orion Star Data/My Folder/PRODUCT_DIM" (Table)
  --> Contains: "Product Name" (InformationMapDataItem)
        --> Is dependent on: "Product_Name" (Column) - in "/Shared Data/Orion Star Data/My Folder/PRODUCT_DIM" (Table)
  --> Contains: "Total Retail Price For This Product" (InformationMapDataItem)
```

Example: Impact Report

The following command produces an impact report for a column with the name Product_Name. This report can be used to identify the objects that would be affected if a change is made to the Product_Name column. The report is written in list format to the file ProductNameImpact.

```
sas-relationship-reporter -host host01.example.com -port 7980 -user domain1\user123 -password pw123
-report impact -types column -name "Product_Name" -file ProductNameImpact
```

Here is the output from this command. Because this is an impact report, the arrows are pointing to the left instead of to the right.

Note: If your deployment is not current with SAS 9.4M3, the phrase **Is dependent on** is used instead of **Impact** to describe the relationships shown in this example.

```
"Product_Name" (Column)
  <-- Impacts: "Product Name" (InformationMapDataItem) - in "/Shared Data/Orion Star Data/Product Orders" (Information map (relational))
      <-- Contains: "/Shared Data/Orion Star Data/Product Orders" (Information map (relational))
          <-- Impacts:: "/Shared Data/Orion Star Data/Sales by Product Group.srx" (SAS report)
  <-- Contains: "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
      <-- Impacts:: "/Shared Data/Orion Star Data/Product Orders" (Information map (relational))
          <-- Impacts:: "/Shared Data/Orion Star Data/Sales by Product Group.srx" (SAS report)
```

Example: Custom Report

The following command produces a custom report that shows, to a depth of 5, the objects that the Sales by Product Group report depends on or is associated with.

```
sas-relationship-reporter -host host01.example.com -port 7980 -user domain1\user123 -password pw123
-rel {-depth 5 -types D A} -types Report -name "Sales by Product Group" -file Custom_Report
```

Here is the output from this command:

```
"/Shared Data/Orion Star Data/My Folder/Sales by Product Group.srx" (SAS report)
  --> Is dependent on: "/Shared Data/Orion Star Data/Product Orders" (Information map (relational))
      --- Is associated with: "SASApp" (Application server)
      --> Is dependent on: "/Shared Data/Orion Star Data/ORDER_FACT" (Table)
          --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
              --- Is associated with: "SASApp" (Application server)
      --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
          --- Is associated with: "SASApp" (Application server)
      --> Is dependent on: "/Shared Data/Orion Star Data/PRODUCT_DIM" (Table)
          --> Is dependent on: "/Shared Data/Orion Star Data/orgold" (Library)
              --- Is associated with: "SASApp" (Application server)
      --> Is dependent on: "/Shared Data/Orion Star Data/Sample: Multiple Output Formats" (Stored process)
          --- Is associated with: "C:\Program Files\SASHome\SASFoundation\9.4\inttech\sample" (Directory)
          --- Is associated with: "SASApp - Logical Workspace Server" (Logical workspace server) - in "SASApp" (Application server)
```

Common Options for the Relationship Reporting Tools

Specifying Web Connection Options

The sas-relationship-loader and sas-relationship-reporter commands require connection options to log on to the SAS Web Server or SAS Web Application Server. These options, which are represented in syntax statements as *web-connection-options*, are as follows:

Table 26.1   Web Connection Options for Batch Relationship Commands

Option	Description
-host *host-name*	Identifies the host machine for the SAS Web Server. If your deployment does not include SAS Web Server, specify the host machine for the SAS Web Application Server.
	Note: You must provide a fully qualified host name (for example, host1.mycompany.com). Do not specify *localhost*.
-port *port*	Specifies the port on which the SAS Web Server runs. If your deployment does not include SAS Web Server, specify the port on which the SAS Web Application Server runs.
-user *user-ID*	Specifies the user ID of the connecting user.
	Note: For sas-relationship-loader, you must specify an unrestricted user (for example, sasadm@saspw). For sas-relationship-reporter, you can specify any authorized user. However, the report output will be limited based on the users' permissions.
-password *password*	Specifies the password of the specified user.
-protocol HTTP | HTTPS	Specifies the communications protocol that is used by the specified host machine and port. If this option is not specified, the default protocol (HTTP) is assumed. If your middle tier has been configured to use Transport Layer Security (TSL), be sure to specify **HTTPS**.

Specifying Object Paths

In the sas-relationship-loader and sas-relationship-reporter commands, *object-paths* specifies the path locations of the objects whose relationship data is to be loaded or reported. You can specify any number of locations. Leave a space between each location. If a location includes spaces, then enclose the location in double quotation marks.

Use the following syntax to specify a location:

/folder-1/folder-2/...folder-n/<object-name(object-type)>

Follow these rules when specifying locations:

- Locations are relative to the **SAS Folders** node. Therefore, the first folder that you specify in a location must be located directly beneath **SAS Folders**.

- Be sure to precede the first folder name with a slash (/).

- If you specify a folder but you do not specify an object name, then all objects and folders beneath that folder are included.

- To specify objects in your personal folder, you must specify the actual path (**/User Folders/***user-name***/My Folder** or **/Users/***user-name***/My Folder**) rather than the shortcut (**/My Folder**). Note that the name of the parent folder for user folders varies depending on the SAS release in which the folders were created.

- If you specify an object name, then you must also specify the object's type. For a list of object types, see "Specifying Object Types" on page 312. Object types are case sensitive.

Here are some examples of locations:

- **"/User Folders/sasdemo/My Folder"**

  This example specifies all objects and folders that are in the personal folder of the user named sasdemo.

- **"/Shared Data/Orion Star Data/Customer Orders(InformationMap)"**.

  This example specifies the Customer Orders information map, which is located in **/Shared Data/Orion Star Data**

- **"/Shared Data/Orion Star Data/Customer Orders(InformationMap)" "/Shared Data/Orion Star Data/ CUSTOMER_DIM(Table)" "/Shared Data/Orion Star Data/ ORDER_FACT(Table)" "/Shared Reports/Orion Star Reports/Sales by Customer Type(Report)"**

  This example specifies the Customer Orders information map, the CUSTOMER_DIM table, and the ORDER_FACT table, all of which are located in **/Shared Data/Orion Star Data**; and the Sales by Customer Type report, which is located in **/Shared Reports/Orion Star Reports**.

Specifying Filtering Arguments

In the sas-relationship-loader and sas-relationship-reporter commands, use the following arguments when specifying *content-filters* or *relationship-content-filters*.

Note: When specifying *relationship-content-filters*, be sure to enclose the argument list in braces ({ and }).

TIP All argument names are case sensitive.

-attrName *resource-attribute-name*
  specifies the name of a resource attribute. Objects that have a resource attribute that exactly matches the specified attribute name and value will be included.

  Only certain types of objects (for example, objects used by SAS Data Integration Studio) support resource attributes.

Valid in	This option is valid in *content-filters* and *relationship-content-filters* for sas-relationship-reporter. This option is not valid in *content-filters* for sas-relationship-loader.

Interaction	You must also use -attrValue to specify a particular value for the attribute.

-attrValue *resource-attribute-value*
   specifies a value for the resource attribute that is specified in -attrName. Objects that have a resource attribute that exactly matches the specified attribute name and value will be included.

Valid in	This option is valid in *content-filters* and *relationship-content-filters* for sas-relationship-reporter. This option is not valid in *content-filters* for sas-relationship-loader.
Interaction	This option is ignored if -attrName is not specified.

-before *from-absolute-date | from-relative-date*
   specifies that objects are to be included only if they were created or modified on or before the specified date. You can specify an absolute date or a date relative to the current date.

Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
Tip	To specify a date range, specify -since along with this option.
See	"Specifying Dates, Times, and Time Periods" on page 311

-created
   specifies that the -since and -before options are to be based on each object's Created date. If -created is specified, then objects are included only if the Created date meets the criteria specified in -since and -before.

Default	If -since or -before is specified but -created is not specified, then -modified is assumed.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-excludeSubTypes
   specifies that subtypes of the specified object types are to be excluded. In the list of object types, include one or more object types that are specified in the -types option.

   For example, suppose you specify the following:

```
-types Report,Job -excludeSubtypes
```

   Objects with the type **Report** and **Job** are included. Objects with the type **Job.Cube**, **Report.Component**, **Report.Image**, and **Report.StoredProcess** are excluded.

-folder *folder-path*
   specifies that objects are to be included only if they are contained in the specified folder within the SAS Folders tree.

-folderTree *folder-path*
   specifies that objects are to be included only if they are contained in the specified folder, or its subfolders, within the SAS Folders tree.

-includeDesc
   specifies that the character string that is specified in -name will be searched for in both the name and the description of each object.

Interactions	This option is ignored if -name is not specified.
	The -nameMatchType option does not apply to the object description. It applies only to the object name. For the object description, the CONTAINS match type is always used.

-keywords *keyword*
    specifies one or more keywords to be used as filtering criteria. If an object has a keyword that exactly matches one or more of the specified keywords, then the object will be included. If a keyword contains spaces, then enclose the value in double quotation marks.

-modified
    specifies that the -since and -before options are to be based on each object's Modified date. If -modified is specified, then objects are included only if the Modified date meets the criteria specified in -since and -before.

Note: If an object has never been modified, then the object's Modified date is equal to its Created date.

Default	If -since or -before is specified, and if you specify neither -created nor -modified, then -modified is assumed.
Interaction	If you specify both -created and -modified, then -modified takes precedence.

-name *character-string*
    specifies a character string that must be present in an object's name in order for the object to be included. All other objects are ignored.

    This option identifies only objects. It does not apply to folders. If the string contains spaces, then enclose the string in double quotation marks.

    The matching of character strings is not case sensitive.

Default	CONTAINS
Interactions	If you specify -includeDesc along with -name, then the object is included if the text string is present in either the object's name or its description.
	You can use the -nameMatchType option to further define how the -name option is to be applied.

-nameMatchType CONTAINS | EQUALS | STARTSWITH
    specifies how the -name option is to be applied. The following values are valid:

CONTAINS	specifies that the object's name must contain the character string that is specified in -name in order for the object to be included.
EQUALS	specifies that object's name must exactly match the character string that is specified in -name in order for the object to be included.
STARTSWITH	specifies that the object's name must begin with the character string that is specified in -name in order for the object to be included.

The matching of character strings is not case sensitive.

Default	CONTAINS
Interactions	This option is ignored if -name is not specified.
	This option applies only to the object name. If you specify -includeDesc, this option does not apply to the description.

-respName *name-of-responsible-user*

specifies a user name that must be present in an object's Responsibilities metadata in order for the object to be included. The specified name must exactly match the user's Name field in the metadata.

Interaction	You can use -respRole to specify whether the user is to be an owner or an administrator for the object.

-respRole OWNER | ADMINISTRATOR

specifies the role of the user that is specified in -respName. The valid values are OWNER and ADMINISTRATOR. Objects will be included only if the object's Responsibilities metadata specifies this role for the user that you specify in -respName.

Interaction	This option is ignored if -respName is not specified.

-since *from-absolute-date | from-relative-date | relative-time-period*

specifies that objects are to be included only if they were created or modified on or after the specified date, or during the specified time period. You can specify an absolute date, a date relative to the current date, or a time period relative to the current date.

Interaction	You can specify either -created or -modified with this option. The default setting is -modified.
See	"Specifying Dates, Times, and Time Periods" on page 311

-types *object-type-1, object-type-2....object-type-n*

specifies a comma-separated list of the types of objects that are to be included. Other types of objects will be ignored.

For a list of object types, see "Specifying Object Types" on page 312. Object types are case sensitive.

By default, if you specify an object type that has one or more subtypes, then the subtypes are included. For example, if you specify **Report**, then **Report.Component**, **Report.Image**, and **Report.StoredProcess** are automatically included.

Interaction	If you want to exclude subtypes of these object types, then use the -excludeSubTypes option with this option.

Using the SAS Metadata Bridge Relationship Loader (sas-metabridge-relationship-loader)

About the SAS Metadata Bridge Relationship Loader

The SAS Metadata Bridge Relationship Loader, new with SAS 9.4M3, enables you to use SAS Metadata Bridges to load third-party lineage information to the relationship database. All objects and relationships that are supported by Meta Integration Model Bridge are available to be loaded.

The command **sas-metabridge-relationship-loader** is located in the following path:

SAS-installation-directory/SASMetadataBridges/version-number/ tools

You can execute this command on any machine where SAS Metadata Bridges has been installed. SAS Metadata Bridges is supported only on Windows.

Syntax for the SAS Metadata Bridge Relationship Loader

sas-metabridge-relationship-loader

> *web-connection-options*
>
> <-bridgeDirectory *bridge-directory* >
>
> -bridgeList <-outputFile *output-path-and-filename*> |
>
> -bridgeInfo -bridgeIdentifier *bridge-identifier* <-outputFile *output-path-and-filename*>
> > <-bridgeOptions *bridge-options-path-and-filename*> |
>
> -loadRelationships -bridgeIdentifier *bridge-identifier* -bridgeOptions *bridge-options-path-and-filename*
> > <-bridgeProfile *bridge-profile-path | bridge-profile-path-and-filename*>
> > <-equivalentRelationships *equivalencies-path-and-file-name*> <-bridgePassword="*bridgeParameterIdentifier*":"*password*">
>
> -clean <-bridgeOptions *bridge-options-path-and-filename*>
>
> -deleteOnMerge

Note: The deleteOnMerge option is available as of SAS 9.4m8.

T I P All command options are case sensitive.

Syntax Description for the SAS Metadata Bridge Relationship Loader

> **web-connection-options**
> specifies connection options to log on to the SAS Web Server or SAS Web Application Server.
>
> **Requirement** Required

Note	For this command, you must specify an unrestricted user (for example, sasadm@saspw) and assign the ROLE_ADMIN role in the SAS Web Administration Console.

See	"Specifying Web Connection Options" on page 363

-bridgeDirectory *bridge-directory*

specifies the location of SAS Metadata Bridges if it is different from the default location.

Default	*SAS-installation-directory*/**SASMetadataBridges**

Requirement	Required only if the directory is different from the default location.

-bridgeIdentifier *bridge-identifier*

specifies the identifier of a particular bridge. When used with -bridgeInfo, specify the bridge for which to get information. When used with -loadRelationships, specify the bridge for which you are loading relationships.

> **TIP** To obtain the identifier, use -bridgeList.

Requirement	Required with -bridgeInfo and -loadRelationships.

-bridgeInfo

requests options and other information for the bridge that is specified in -bridgeIdentifier. The detailed information includes documentation of the options, including what is required and enumerated values. An options template is created if you also specify -bridgeOptions.

> **TIP** As best practice, always use this option along with -outputFile to obtain the bridge-specific template file and documentation before loading relationships. For an example, see "Step 3: Obtain Bridge Information and an Options Template" on page 371.

Requirement	You must specify one of the following: -bridgeInfo, -bridgeList, -loadRelationships, or -clean.

-bridgeList

requests a list of the available licensed bridges and their bridge identifiers. To write the list to a file, specify -outputFile.

> **TIP** Use this option to obtain the bridge identifier to specify with -bridgeInfo or -loadRelationships. For an example, see "Step 1: Obtain a List of Available Bridges" on page 371.

Requirement	You must specify one of the following: -bridgeInfo, -bridgeList, -loadRelationships, or -clean.

-bridgeOptions *bridge-options-path-and-filename*

specifies the path and name of the file that contains bridge options for a particular bridge. Bridge options are unique for each bridge.

Interactions	When used with -bridgeInfo, specify the path and filename for the options template to be created. For an example, see "Step 3: Obtain Bridge Information and an Options Template" on page 371. You can then edit the template as illustrated in "Step 4: Review the Bridge Information and Edit the Options Template" on page 372. The file can then be used as input to -loadRelationships and -clean.

When used with -loadRelationships, specify the path and filename of a valid bridge options file to be used as input to the relationship load. For an example, see "Step 5: Load the Relationships" on page 372.

When used with -clean, specify the path and filename of a valid bridge options file that specifies parameters for the delete.

-bridgeProfile *bridge-profile-path | bridge-profile-path-and-filename*
   specifies the absolute path (or the absolute path and filename) for the bridge profile, which is used to determine which relationships to load. Specify this option with -loadRelationships if you want to use a profile other than the default.

   Default   `SAS-installation-directory/SASMetadataBridges/`
   `version-number/conf/MIRProfiles/Meta Integration.xml`

-deleteOnMerge
   if metadata was deleted in the source after the first import, specifies to delete the same metadata in the relationships service on the second import.

Note: The deleteOnMerge option is available as of SAS 9.4m8.

-equivalentRelationships *equivalencies-path-and-filename*
   specifies the path and name of a file that specifies the equivalent SAS relationships to create during relationship loading.

   In the file, you identify imported third-party tables that are the same or equivalent to tables that are modeled in SAS. Using this information, the relationship tools can report lineage from SAS objects to imported metadata.

   Interaction   This is an optional input to -loadRelationships.

   See   "XML Specification for Defining Equivalent Relationships" on page 372 and "Example: Creating and Using an Equivalent Relationships File" on page 373.

-loadRelationships
   requests that relationships be loaded to the relationship database from a third-party source. A valid bridge options file must be specified in -bridgeOptions.

   T I P   Before loading relationships, use -bridgeInfo (along with -bridgeOptions and -outputFile) to obtain a bridge-specific options template file and documentation on how to specify the source and the options for the bridge. Then edit the template as illustrated in "Step 4: Review the Bridge Information and Edit the Options Template" on page 372.

   Requirement   You must specify one of the following: -bridgeInfo, -bridgeList, -loadRelationships, or -clean.

-outputFile *output-path-and-filename*
   specifies the path and filename for command output. When used with -bridgeList, specify the file that is to contain the list of supported bridges. The file is written in XML format.

   Interactions   When used with -bridgeInfo, specify the file that is to contain detailed bridge information (including documentation for specifying options). A plain text file is created.

When used with -bridgeList, specify the file that is to contain the list of supported bridges. An XML file is created.

-clean

deletes from the relationship database the resources and relationships for a particular third-party source. The third-party source is included in the options template file that is specified in -bridgeOptions.

-bridgePassword="*bridgeParameterIdentifier:password* **"|**
"*bridgeParameterIdentifier1:password1,bridgeParameterIdentifier2:password2***"**

specifies the bridgeParameterIdentifier and password value that are specified in the bridgeOptions file. If multiple passwords are specified in the bridgeOptions file, then you must separate them with commas.

TIP Use -bridgePassword to specify the password if you cannot explicitly include the password in the options file.

Interaction This is an optional input to -loadRelationships.

Example: Loading Third-party Lineage Information to the Relationship Database

This example provides step-by-step instructions for loading third-party lineage information to the relationship database. The example uses the CA ERwin 4.x Data Modeler bridge.

Step 1: Obtain a List of Available Bridges

Use the following command to obtain a list of licensed bridges and their bridge identifiers and write them to an XML file.

Note: 7980 is the default port number for the SAS Web Application Server. The port number might be different in your deployment.

```
sas-metabridge-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
-bridgeList -outputFile "c:\temp\bridgeList.xml"
```

Step 2: Determine the Bridge Identifier

Open the **bridgeList.xml** file that was created in step 1, and find the bridge identifier of interest. The following entry indicates that the bridge identifier for the CA ERwin Data Modeler v4.x bridge is **CaErwin4Xml**.

```
<Bridge bridgeDisplayName="CA ERwin 4.x Data Modeler" bridgeIdentifier="CaErwin4Xml"/>
```

Step 3: Obtain Bridge Information and an Options Template

Using the bridge identifier that you obtained in step 2 (**CaErwin4Xml**), use the following command to obtain an options template and other information for the CA Erwin Data Modeler v4.x bridge.

A template containing the available options for the bridge is written to **CaErwin4XmlOptions.xml**. Other bridge information is written to **CaErwin4XmlInfo.txt**.

```
sas-metabridge-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
-bridgeIdentifier CaErwin4Xml -bridgeOptions "c:\temp\CaErwin4XmlOptions.xml"
-bridgeInfo -outputFile "c:\temp\CaErwin4XmlInfo.txt"
```

Step 4: Review the Bridge Information and Edit the Options Template

1. Review the bridge information file (**CaErwin4XmlInfo.txt**) that was created in step 3.

   TIP It is important to always review this file because it might contain additional steps, tips, and information about using the bridge. For example, the file for the CA Erwin Data Modeler v4.x bridge provides instructions for using the data modeling tool to save the model in XML format.

2. Open the options template file (**CaErwin4XmlOptions.xml**) that was created in step 3, and modify it according to the documentation in the bridge information file.

   For example, this options template file contains the following XML element in which to specify the location and name of the file to be created by the data modeling tool:

```
<BridgeParameter bridgeParameterIdentifier="File" displayName="File" mandatory="true"/>
```

   The following modifications direct the CA Erwin Data Modeler tool to save the model in **c:\temp\MyCaErwin4DataModel.xml**:

```
<BridgeParameter bridgeParameterIdentifier="File" displayName="File" mandatory="true">
c:\temp\MyCaErwin4DataModel.xml
</BridgeParameter>
```

   TIP The steps needed to edit the options template vary from bridge to bridge. Be sure to follow the instructions that are provided in your bridge information file.

   Note: If your options template requires a password value, but you cannot explicitly include the password in the options file, you can use the -bridgePassword option with the -loadRelationships option to specify the password. If you use the -bridgePassword option to specify the password, then there is no requirement to explicitly include the password in the options file.

   Save the modified options template file.

Step 5: Load the Relationships

Use the following command to export relationships from the source that is specified in the bridge options file (**CaErwin4XmlOptions.xml**). The file MyCaErwin4DataModel.xml is profiled for lineage and other relationship information, and this information is loaded to the relationship database.

```
sas-metabridge-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw -password pw123
-bridgeIdentifier CaErwin4Xml -bridgeOptions c:\temp\CaErwin4XmlOptions.xml
-loadRelationships
```

   TIP As part of this step, you can also specify an equivalent relationships file. See "XML Specification for Defining Equivalent Relationships" on page 372 and "Example: Creating and Using an Equivalent Relationships File" on page 373.

XML Specification for Defining Equivalent Relationships

When loading relationships with sas-metabridge-relationship-loader, you can use the -equivalentRelationships option to specify an equivalent relationships file. In this file, you identify imported third-party libraries and tables that are the same or equivalent to libraries and tables that are modeled in SAS. The relationship reporting tools can then report lineage from SAS objects to imported metadata. Use the following syntax to create the equivalent relationships file.

XML Syntax for Defining Equivalent Relationships

```
<EquivalentRelationships>

    <Equivalent type="Library | Table">

        <ImportedResource name="schema-name" /> |

        <ImportedResource name="schema-name.table-name"/>

        <Resource objectID="object-id" objectTypeName="Library | Table"/> |

        <Resource metadataPath="metadata-location"/>

    </Equivalent>
</EquivalentRelationships>
```

Syntax Description for Defining Equivalent Relationships

type="*object-type*"

specifies the object type of the third-party resource. You must specify either **Library** or **Table**.

> **T I P** Object types are case sensitive.

name="*schema-name*" | name="*schema-name.table-name*"

specifies the name of imported third-party resource. For libraries, specify the schema name. Specify table names in the format *schema-name.table-name*.

objectID="*object-id*"

specifies the object ID that is assigned to the resource in SAS metadata.

> **Interaction** The objectTypeName attribute must be specified with this attribute.

objectTypeName="Library | Table"

specifies the object type of the resource in SAS metadata. You must specify either **Library** or **Table**. Object types are case sensitive.

> **Interaction** The objectID attribute must be specified with this attribute .

metadataPath="*metadata-location*"

identifies the location of the object in SAS metadata relative to the **SAS Folders** node. This parameter can be specified instead of objectID and objectTypeName.

Use the following syntax: */folder-1/folder-2/...folder-n/object-name*(Library | Table). For more details, see "Specifying Object Paths" on page 363.

Example: Creating and Using an Equivalent Relationships File

This example provides step-by-step instructions for creating an equivalent relationships file and loading equivalent relationships to the relationships database.

1. If you have not already done so, create a library by using the **Data Library Manager** plug-in for SAS Management Console. Then use the **Register Tables** option to import the metadata for the third-party relational table into the library. Note the schema name of the external database (ORSTAR in this example) for later use.

2. Create the equivalent relationships file. Refer to "XML Specification for Defining Equivalent Relationships" on page 372 and to the following example:

```
<EquivalentRelationships>

  <Equivalent type="Library">
   a <ImportedResource name="ORSTAR"/>
```

```
    b <Resource metadataPath="/Shared Data/OrionStarErwin/Libraries/
        OrionStarErwin(Library)"/>
      </Equivalent>

      <Equivalent type="Table">
    c <ImportedResource name="ORSTAR.Customer_Dim"/>
    d <Resource metadataPath="/Shared Data/OrionStarErwin/Data/Customer_Dim(Table)"/>
      </Equivalent>

      <Equivalent type="Table">
        <ImportedResource name="ORSTAR.Time_Dim"/>
        <Resource metadataPath="/Shared Data/OrionStarErwin/Data/Time_Dim(Table)"/>
      </Equivalent>

      <Equivalent type="Table">
        <ImportedResource name="ORSTAR.Product_Dim"/>
        <Resource metadataPath="/Shared Data/OrionStarErwin/Data/Product_Dim(Table)"/>
      </Equivalent>

    </EquivalentRelationships>
```

1 Specify the schema name for the external database (**ORSTAR** in this example).

2 Specify the metadata path, library name, and object type of the SAS metadata library that is equivalent to the third-party resource specified in the preceding element. In this example, the path is **/Shared Data/OrionStarErwin/ Libraries/**, the library name is **OrionStarErwin**, and the object type is **Library**.

3 Specify the schema name for the external database and the table name of the third-party table. In this example, the schema name is **ORSTAR** and the table name is **Customer_Dim**.

4 Specify the metadata path, table name, and object type of the SAS table that is equivalent to the third-party table specified in the preceding element. In this example, the path is **/Shared Data/OrionStarErwin/Data/**, the table name is **Customer_Dim**, and the object type is **Table**.

Add an **<Equivalent type="Table">** element for each additional table.

3. After creating the equivalent relationships file, load the third-party lineage information to the relationships database as described in "Example: Loading Third-party Lineage Information to the Relationship Database" on page 371. In "Step 5: Load the Relationships" on page 372, use the -equivalentRelationships option to specify your equivalent relationship file, as shown in the following example:

```
sas-metabridge-relationship-loader -host host01.example.com -port 7980 -user sasadm@saspw-password pw123
-bridgeIdentifier CaErwin4Xml -bridgeOptions c:\temp\CaErwin4XmlOptions.xml -loadRelationships
-equivalentRelationships c:\temp\EquivalentRelationships.xml
```

Chapter 27

Using the Batch Tools for Metadata Server Administration

About the Batch Tools for Metadata Server Administration

The following batch tools are provided for metadata administration:

Tool	Purpose	Documentation Location
sas-backup-metadata	Run an unscheduled (ad hoc) backup of the metadata server. You can also list metadata server backups and recoveries that are present in backup and recovery history.	"Back Up Metadata (sas-backup-metadata)" on page 376
sas-recover-metadata	Restore the metadata server from a backup. You can also list metadata server backups and recoveries that are present in backup and recovery history.	"Recover Metadata (sas-recover-metadata)" on page 377

Tool	Purpose	Documentation Location
sas-create-repository	Create and initialize a new metadata repository.	"Create a Repository (sas-create-repository)" on page 379
sas-create-project-repository	Create and initialize a new project repository.	"Create a Project Repository (sas-create-project-repository)" on page 380
sas-delete-repository	Delete a foundation, custom, or project repository.	"Delete a Repository (sas-delete-repository)" on page 380
sas-update-metadata-profile	Update the metadata profiles in your SAS Application Server's configuration with information about a metadata server cluster.	"Update Metadata Profiles (sas-update-metadata-profile)" on page 381
sas-analyze-metadata	Run selected tests on metadata and repair identified problems.	"Analyze and Repair Metadata (sas-analyze-metadata)" on page 382

These tools are located in **`SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin`**. Before using the tools, refer to the following topics:

- "General Instructions for Using the SAS Intelligence Platform Batch Tools" on page 310

- "Common Options" on page 313

Note: The **`/tools/admin`** directory also contains the following commands:

- commands that are part of the Deployment Backup and Recovery tool. For releases prior to SAS 9.4M5, these commands include sas-add-backup-customdir, sas-backup, sas-display-backup, sas-display-backup-source-content, sas-display-backup-config, sas-display-backup-source-config, sas-display-backup-customdir, sas-display-recovery, sas-list-backup-schedule, sas-list-backups, sas-recover-offline, sas-remove-backup-customdir, sas-remove-backup-schedule, sas-set-backup-schedule, sas-status-backup, and sas-update-backup-config. For SAS 9.4M5, the following commands are no longer included: sas-add-backup-customdir, sas-display-backup-customdir, sas-remove-backup-customdir, and sas-display-backup-source-config. For information about these commands, see Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

- the commands sas-port-metadata and sas-upgrade-metadata. These commands are not included in the documentation. You should use these commands only if you are directed to do so by SAS Technical Support.

Back Up Metadata (sas-backup-metadata)

Use sas-backup-metadata to run an unscheduled (ad hoc) backup of the metadata server. The tool backs up the server to the destination that is currently configured in the metadata server backup facility. You can also use this tool to list backups and recoveries that are present in backup and recovery history.

The syntax is as follows:

sas-backup-metadata

> *connection-options*
> <-reorg>
> <-comment *comment*>
> <-list>
> <-since *from-absolute-date | from-relative-date|relative-time-period*>
> <-before *to-absolute-date | to-relative-date*>

The following list provides details:

connection-options
> See "Common Options" on page 313.

-reorg
> reorganizes repository files during the backup to reclaim unused disk space that was used by previously deleted metadata objects. If you specify this option, the backup facility pauses the server to a READONLY state while the reorganization is taking place. This option cannot be specified with -list.

-comment *comment*
> specifies a comment that is to be stored in the backup history with this backup. This option cannot be specified with -list.

-list
> displays a list of backups and recoveries that exist in the current backup and recovery history. If you specify this option, a backup is not performed.

-since *from-absolute-date | from-relative-date | relative-time-period*
> specifies that backups and recoveries are to be listed only if they occurred on or after the specified date, or during the specified time period. This option can be specified only with -list. You can specify an absolute date, a date relative to the current date, or a time period relative to the current date.

> **See** "Specifying Dates, Times, and Time Periods" on page 311

-before *from-absolute-date | from-relative-date*
> specifies that backups and recoveries are to be listed only if they were created or modified on or before the specified date. This option can be specified only with -list. You can specify either an absolute date or a date relative to the current date.

> **Tip** To specify a date range, specify -since along with this option.

> **See** "Specifying Dates, Times, and Time Periods" on page 311

See Also

- Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153
- Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169

Recover Metadata (sas-recover-metadata)

Use sas-recover-metadata to restore the metadata server from a backup. The syntax is as follows:

sas-recover-metadata

> *connection-options*
>   *backup-name*
>   -time *date | date-and-time*
>   <-comment *comment*>
>   <-rollForward>
>   <-includeConfig>
>   <-ignoreVerify>
>   <-list>

The following list provides details:

connection-options
See "Common Options" on page 313.

backup-name
The name of the backup from which the metadata is to be recovered (for example, 2013–002–22T01_00_00_055_00).

-time *date | date-and-time*
specifies a date in the format *yyyy-mm-dd*, or a date and time in the format *yyyy-mm-dd*T*hh:mm<:ss>*. This date and time are used as follows:

- If *backup-name* is not specified, the recovery uses the backup that most closely precedes the specified date and time.

- If -rollForward is specified, then the recovery also includes journal transactions that were recorded after the backup was created through the specified date and time.

- If -list is specified, then the name of the backup that most closely precedes the specified date and time is displayed. No recovery is performed.

-comment *comment*
specifies a comment that is to be stored in the recovery history for this recovery. This option cannot be specified with -list.

-rollForward
specifies that the recovery is to include transactions that were recorded in the metadata server journal file after the backup was created. The recovery includes journal transactions through the date and time that are specified in the -time option.

This option cannot be specified with -list.

-includeConfig
This option does the following:

- deletes all files from the server's configuration directory (for example, **SAS-configuration-directory/Lev1/SASMeta/MetadataServer**)

- from the **ConfigurationFiles** subdirectory in the backup location, copies all files (with the exception of MetadataServerBackupConfiguration.xml, MetadataServerBackupHistory.xml, and MetadataServerRecoveryManifest.xml) to the server's configuration directory

CAUTION:
If you specify -includeConfig, you will lose any additions or modifications to files in the MetadataServer directory that have been made since the backup was taken. As a best practice, you should specify this option only if your configuration files have become corrupted.

This option cannot be specified with -list.

-ignoreVerify
>   specifies that the recovery is to be executed without verifying that the backup files are correct. If this option is not specified, the verification step prevents the recovery from executing if it detects potential problems with the backup files (such as incorrect file names, file sizes, or universally unique identifiers).
>
>   This option is ignored if it is specified with -list.

-list
>   displays a list of backups and recoveries that exist in the current backup and recovery history. If you specify this option, a recovery is not performed.

See Also

- Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153
- Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169

Create a Repository (sas-create-repository)

Use sas-create-repository to create and initialize a new metadata repository. The repository can be either a custom repository or a foundation repository. The syntax is as follows:

sas-create-repository

>   *connection-options*
>   *repository-name*
>   <-path *repository-path*>
>   <-userRoot *path-to-user-folders*>
>   <-desc *repository-description*>

The following list provides details:

connection-options
>   See "Common Options" on page 313.

repository-name
>   specifies the name of the new repository.

-path *repository-path*
>   specifies the location where the repository is to be stored.
>
>   If this option is not specified, the repository files are created in the following location in the file system of the metadata server:
>
>   **SAS-configuration-directory/Lev1/SASMeta/MetadataServer/
>   MetadataRepositories/repository-name**

-userRoot *path-to-user-folders*
>   specifies the location in metadata where user folders are created. This option is valid only for foundation repositories.

-desc *repository-description*
>   specifies optional descriptive text for the repository.

Create a Project Repository (sas-create-project-repository)

Use sas-create-project-repository to create and initialize a new project repository. The syntax is as follows:

sas-create-project-repository

> *connection-options*
> *repository-name*
> -owner *user-name*
> <-path *repository-path*>
> <-desc *repository-description*>

The following list provides details:

connection-options
> See "Common Options" on page 313.

repository-name
> specifies the name of the new repository.

-owner *user-name*
> specifies the identity name of the user for whom the repository is being created. This option is required.

-path *repository-path*
> specifies the location where the repository is to be stored.
>
> If this option is not specified, the repository files are created in the following location in the file system of the metadata server:
>
> **SAS-configuration-directory/Lev1/SASMeta/MetadataServer/**
> **MetadataRepositories/repository-name**

-desc *repository-description*
> specifies optional descriptive text for the repository.

Delete a Repository (sas-delete-repository)

Use sas-delete-repository to delete a foundation, custom, or project repository. The syntax is as follows:

sas-delete-repository *connection-options repository-name*

The following list provides details:

connection-options
> See "Common Options" on page 313.

repository-name
> specifies the name of the repository that is to be deleted.

Update Metadata Profiles (sas-update-metadata-profile)

If you have configured a metadata server cluster, use sas-update-metadata-profile to update the metadata profiles in your SAS Application Server's configuration with information about the cluster. You must run this command at the following times:

- after completing the initial configuration of the cluster

- after making changes to the cluster configuration (including adding or removing nodes)

The command should be run on each machine that contains an object spawner, a SAS/CONNECT spawner, or components of SAS Application Servers (such as workspace servers, pooled workspace servers, OLAP servers, and stored process servers).

You can also use this command to create a new profile that contains the cluster information.

The syntax is as follows:

sas-update-metadata-profile

> *connection-options*
> -updateAll *path* | -update *path-and-filename* | -create *path-and-filename*
> <-profileUser *profile-user*>
> <-profilePassword *profile-password*>

The following list provides details:

connection-options
> See "Common Options" on page 313.

Specify one of the following actions:

-updateAll *directory-name*
> specifies that metadata profiles in all files named metadataConfig.xml and metaProfile.xml within the named directory (or in its immediate subdirectories) are to be updated with information from the metadata server's cluster definition. When the updates are made, the original files are saved and renamed to *filenname_*old*n*.xml (for example, metaConfig_old1.xml).

> *Note:* By default, each of the following directories includes a metadata profile called metadataConfig.xml:

> - **SAS-configuration-directory/levn**

> - **SAS-configuration-directory/levn/ObjectSpawner**

> - **SAS-configuration-directory/levn/ConnectSpawner**

> **TIP** To ensure that the metadata profiles on each SAS Application Server machine have the latest metadata server cluster information, run the following command on each of the machines:

```
sas-update-metadata-profile connection-options -updateAll
   SAS-configuration-directory/levn
```

-update *path-and-filename*
> specifies that the metadata profile in the specified file, which is located in the specified path, is to be updated with information from the metadata server's cluster definition. Files that contain metadata profiles are generally named metadataConfig.xml or metaProfile.xml.
>
> When the updates are made, the original file is saved and renamed to *filenname_oldn*.xml (for example, metaConfig_old1.xml).

-create *path-and-filename*
> specifies that a metadata profile containing information from the metadata server's cluster definition is to be created in a new file with the specified name within the specified directory location. The directory must already exist in the file system.

You can specify the following options with any of the previously listed actions:

-profileUser *profile-user*
> specifies a user name that is to be specified in the profile. This user name is used to connect to the metadata server when code is submitted to the SAS Application Server. Use this option only if you do not want user credentials to be required in submitted code.

-profilePassword *profile-password*
> specifies the password of the user that is specified in -profileUser. The password is required if -profileUser is specified.

-profileRepository *repository-name*
> specifies the name of the metadata repository to which the SAS Application Server is to connect.

See Also

> Chapter 16, "Using Metadata Server Clustering," on page 211

Analyze and Repair Metadata (sas-analyze-metadata)

About the Analyze and Repair Metadata Batch Tool

> *Note:* If you are running a non-clustered metadata server, the Metadata Analyze and Repair Tools can be run without stopping any SAS servers. For clustered metadata servers, the only Metadata Analyze and Repair tool that can run while in quorum is "Cluster Synchronization". The other Metadata Analyze and Repair tools for clustered metadata servers must be run on a single node that has been started without clustering (-STARTNOCLUSTER option).

Use the Analyze and Repair Metadata Batch Tool (sas-analyze-metadata) to run selected tests on metadata to locate common problems. When possible, you can also use this command to repair problems that the analysis has identified.

This tool is new with SAS 9.4M2. It provides a batch method for running the analyze and repair tools that are available in the **Metadata Manager** node in SAS Management Console. (See "Analyzing and Repairing Metadata" on page 236.)

When you execute sas-analyze-metadata, the metadata server is automatically paused to ADMINISTRATION mode. Therefore, you might want to run the command during

times of low metadata server activity. The ADMINISTRATION state prevents metadata changes from occurring while the analysis process is running, except that unrestricted users can continue to change metadata during this time. The server is automatically resumed when the analysis and repair process is completed.

As a best practice, you should back up the metadata server before using this command.

Note: For clustered metadata servers, in most cases you should run sas-analyze-metadata on a single node that has been started without clustering. An exception is the `Synch` tool, which works only on a server that has been started with clustering.

Syntax for the Analyze and Repair Metadata Batch Tool

sas-analyze-metadata

connection-options
    <-tool *tool-1+tool-2+...tool-n*> | <-group *group-1+group-2+...group-n*>
    <-repair>
    <*repository-1 repository-2...repository-n* >

Syntax Description for the Analyze and Repair Metadata Batch Tool

connection-options
    See "Common Options" on page 313.

-tool *tool-1+tool-2+...tool-n*
    specifies one or more analysis and repair tools that are to be executed. Use either plus signs or commas to separate multiple tool names.

    This option cannot be specified with -group. If you omit both -tool and -group, all of the tools are executed.

    The following tool names are valid:

Tool Name	Full Name	Description
Files	Verify Metadata Files	Analyzes key metadata server files to determine whether they are corrupted and, when possible, recommends repairs that can be applied. This tool might be useful if a metadata server failure occurs or if a user experiences unexpected results when accessing the server.
Assoc	Verify Associations	Checks the metadata repository for associations in which one or the other associated object does not exist. This tool might be useful if a user experiences unexpected results when accessing the server.
AuthDomains	Verify Authentication Domains	Checks authentication domain objects to ensure that the object names are valid and unique. It also checks to ensure the valid usage of these objects. When possible, the tool recommends repairs that can be applied.
Permissions	Verify Permissions	Verifies that permission objects exist only in the Foundation repository and recommends repairs that can be applied.
FoldStruct	Validate SAS Folders Structure	Analyzes the integrity of objects contained in the SAS Folders tree and, when possible, recommends repairs that can be applied. This tool identifies issues such as duplicate, missing, or invalid object names; objects that are present in more than one folder; and objects that are stored in a different repository than their parent folders.

Tool Name	Full Name	Description
`OrphAce`	Orphaned Access Controls	Locates metadata objects that are no longer being referenced. Metadata can become orphaned when an application deletes a metadata object but does not delete metadata that is associated with the deleted object. Orphaned objects are candidates for removal by the repair tool.
`OrphProp`	Orphaned Property Metadata	
`OrphTxtStr`	Orphaned Source Code	
`OrphJob`	Orphaned Job Metadata	
`OrphDir`	Orphaned Directories	
`Synch`	Metadata Server Cluster Synchronization	Verifies that metadata is synchronized among all the nodes of a metadata server cluster. If metadata differences are found, the repair option sets a flag on the server, writes messages about the issue to the Analyze and Repair log file, and generates an alert email. The next time the problem node is restarted, the flag causes the node's metadata to be synchronized with metadata on the controller node.
		This tool is new with SAS 9.4M3. No analysis is performed if this tool is run on a non-clustered metadata server or on a clustered server that contains only one node.

For more details about these tools, including best practices, see "Analyzing and Repairing Metadata" on page 236.

-group *group-1+group-2+...group-n*

specifies one or more groups of analysis and repair tools that are to be executed. Use either plus signs or commas to separate multiple group names.

This option cannot be specified with -tool. If you omit both -tool and -group, all of the tools are executed.

The following group names are valid:

Group Name	Tools Contained in the Group
`Physical`	Files, Assoc, Synch
`Security`	AuthDomains, Permissions
`Usage`	FoldStruct, OrphAce, OrphProp, OrphTxtStr, OrphJob, OrphDir

-repair

immediately repairs any problems that are identified by the executed tools.

CAUTION:

Be sure to back up the metadata server before you use the -repair option.
See "Using SAS Management Console to Run an Ad Hoc Metadata Server Backup" on page 176.

> `T I P` When you run the Files tool or the Assoc tool, you should omit the -repair option in the first execution, and review the output to determine whether any repairs are recommended. You can then specify the -repair option in a subsequent execution. However, if the Files tool reports errors (as compared to warnings), call SAS Technical Support instead of using the -repair option.

repository-1 repository-2...repository-n
    specifies one or more repositories to be analyzed. Leave a space between each repository name. If no repositories are specified, then all of the repositories are analyzed.

Examples for the Analyze and Repair Metadata Batch Tool

The following command uses all of the analysis tools to analyze metadata in all repositories, but does not perform any repairs:

```
sas-analyze-metadata –profile "My Server"
```

The following command analyzes the SAS Folders structure in the Custom1 repository:

```
sas-analyze-metadata –profile "My Server" -tool FoldStruct Custom1
```

The following command analyzes metadata files, associations, authentication domains, and permissions in the Foundation repository and repairs any problems:

```
sas-analyze-metadata –profile "My Server" -group Physical+Security
   -repair Foundation
```

Chapter 28
Using the Deployment Backup and Recovery Tool

About the Deployment Backup and Recovery Tool

What Is a Deployment?

A SAS Intelligence Platform *deployment* is an instance of operational SAS software and related components.

CAUTION:
  The Deployment Backup and Recovery tool does not back up your entire deployment. The Deployment Backup and Recovery tool backs up only a subset of the components in your deployment. For more information, see "What is Backed Up by the Deployment Backup and Recovery Tool?" on page 389.

Purpose and Scope

Overview

The Deployment Backup and Recovery tool, new with SAS 9.4, provides an integrated method for synchronizing the backup and recovery of SAS content across multiple tiers and machines in your SAS deployment. The tool is installed and configured on the middle tier as part of the SAS Web Infrastructure Platform and executes commands using the SAS Deployment Agent on each middle-tier and server-tier machine. The tool automatically configures the backup by discovering each of the tiers in your deployment and the components configured on each tier.

Note: Be aware of these limitations of the Deployment Backup and Recovery tool.

- You cannot recover from backup if the data server is unable to start due to database corruption.

- You cannot use the tool to recover a backup from one deployment to another.

- The tool is not a substitute for file system backups and is not recommended for the purpose of backing up system-level files.

- By default, the Deployment Backup and Recovery tool performs a backup once a week (on Sunday morning). It backs up content, not your deployment. For more information about what is and is not backed up, see "What is Backed Up by the Deployment Backup and Recovery Tool?" on page 389 and "What is Not Backed Up by the Deployment Backup and Recovery Tool?" on page 389.

What is Backed Up by the Deployment Backup and Recovery Tool?

The following components are included in the backup:

- SAS Metadata Server: All registered repositories, the repository manager, and configuration files for the SAS Metadata Server are backed up.

- Databases: Databases managed by the SAS Web Infrastructure Platform Data Server (including SAS solution data servers configured with the SAS Deployment Wizard) are backed up.

  Note: If you configured SAS Web Infrastructure Platform Services using an alternative database (that is, a database from a third-party vendor), then it is not backed up with the SAS Deployment Backup and Recovery tool. For more information, see the information about alternative databases in "What is Not Backed Up by the Deployment Backup and Recovery Tool?" on page 389.

- SAS Content Server: The SAS Content Server repository and digital content stored in SharedServices database tables for SAS Content Server are backed up.

  Note: If you configured SAS Web Infrastructure Platform Services using an alternative database (that is, a database from a third-party vendor), then it is not backed up with the SAS Deployment Backup and Recovery tool. For more information, see the information about alternative databases in "What is Not Backed Up by the Deployment Backup and Recovery Tool?" on page 389.

- Configuration directories: The following content in the SAS configuration directories is backed up:

  - the contents of the **Data** directories and **SASEnvironment** directories for the SAS Metadata Server and SAS Application Server context.

  - the directories for SAS Application Server components on the SAS server tier, except batch server components and the SAS Grid Server component.

  - the directories for each spawner used by the SAS Application Server component on the SAS server tier.

- Custom directories: Additional directories are backed up in the **SAS-configuration-directory/Levn** directory if custom directories are configured for the tool.

  Note: Adding the entire Levn configuration directory as a custom directory is not supported. Only adding directories within Levn is supported.

Note: For all of the directories listed here, if symbolic links in these directories point to other locations, the referenced locations are not backed up.

What is Not Backed Up by the Deployment Backup and Recovery Tool?

The Deployment Backup and Recovery tool does not perform a backup of the entire SAS deployment.

The following components are not backed up by the Deployment Backup and Recovery tool:

- The SAS installation files are not backed up.

- The complete contents of the SAS configuration directories are not backed up.

  TIP Adding the entire Levn configuration directory as a custom directory is not supported. Only adding directories within Levn is supported. To back up additional subdirectories directories under ***SAS-configuration-directory*/Levn**, use the command sas-update-backup-config to add them as custom directories. If your deployment is not current with the third maintenance release, then use the command sas-add-backup-customdir on page 426.

- SAS data files that contain source data to SAS applications or solutions are not backed up.

- If symbolic links in these directories point to other locations, the referenced locations are not backed up.

- Batch server components and the SAS Grid Server component are not backed up.

- Alternative databases (that is, databases from third-party vendors) are not backed up:

  - SAS Web Infrastructure Platform Services configures the SAS Web Infrastructure Platform Data Server by default. If you choose to configure an alternative database, the Deployment Backup and Recovery tool cannot be used to back up or restore that database.

  - SAS Web Infrastructure Platform Services configures a default SharedServices database instance in the SAS Web Infrastructure Platform Data Server. If you choose to configure an alternative database for SharedServices, the Deployment Backup and Recovery tool cannot be used to restore the SAS Content Server from backup.

Where Are Backups Stored?

By default, backup files are stored locally on the same machine where the backup source is located. All backup sources are backed up to the following path on their respective host machines: ***SAS-configuration-directory*/Lev1/Backup/Vault**. This directory is created on each machine the first time a backup is executed.

You can also specify a central, network-accessible vault location to store backups. If a central vault is specified, backups are automatically copied from the various host machines to the central vault following each backup operation.

A central vault location is highly recommended to avoid the loss of backup files in the event that a host machine fails.

About SAS Metadata Server Backups

For metadata server backups, the Deployment Backup and Recovery tool runs the metadata server backup utility, which is described in Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169.

If a central vault is not configured, the backup files are copied to ***SAS-configuration-directory*/Lev1/Backup/Vault** on the metadata server machine. If metadata server clustering is configured, the files are copied to the initially configured metadata server.

However, if a central vault is configured, the backup files are copied to the central vault location instead. A central vault is highly recommended if metadata server clustering is configured.

User Credentials for Performing Backups

You must use the SAS Administrator user to perform backup and recovery operations. By default, backups on Windows systems are performed by the Local System account for the SAS Deployment Agent. On UNIX, backups are performed by the SAS Installer user for each server and middle-tier machine.

Ensure that the **`SAS-configuration-directory/Lev1/SASApp/ PooledWorkspaceServer/Logs`** directory and the log files within have Read and Write permissions for the file owners and their primary group.

A special user account to perform backups must be defined in the following situations:

- if you have specified a central vault location and your environment includes one or more Windows hosts

- if a clustered metadata server has been configured and your environment includes one or more Windows hosts

For details, see "Specifying a Backup User to Access a Central Vault Location or a Clustered Metadata Server (Windows Only)" on page 397.

Using the Deployment Backup and Recovery Tool in Heterogeneous Environments

If your deployment has a heterogeneous operating system environment, you can use the Deployment Backup and Recovery tool from SAS 9.4M1. A heterogeneous environment is one in which the host machines that are included in the backup are in different operating system families (for example, both UNIX and Windows).

About SAS Backup Manager

SAS 9.4M3 includes SAS Backup Manager, an easy-to-use interface for the Deployment Backup and Recovery tool. You can use SAS Backup Manager for the following tasks:

- view backup and recovery history

- run an immediate (ad hoc) backup

- view the backup configuration

- modify the backup configuration (except backup filters and custom directories)

- view information about backup and recovery sources

- view and modify the backup schedule

In previous SAS 9.4 releases, these functions were available only through batch commands.

SAS Backup Manager can be accessed from the **Administration** tab of SAS Environment Manager. For more information, see the *SAS Environment Manager 2.5 Administration: User's Guide*, *SAS Environment Manager 2.6 Administration: Fundamentals*, and the Help for SAS Backup Manager.

See Also

- Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153

- Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169

Setting Up Your Backups

Perform the following steps to set up backups in a newly installed or upgraded deployment of SAS 9.4:

Review the Backup Schedule and Make Any Needed Changes

By default, the Deployment Backup and Recovery tool performs a backup automatically each Sunday at 1:00 a.m. Review the schedule and make any changes that are appropriate for your deployment. For example, you might want to schedule backups to occur daily instead of weekly.

To view the current schedule, you can use SAS Backup Manager or the command sas-list-backup-schedule.

To modify the schedule, you can use SAS Backup Manager or the command sas-set-backup-schedule.

T I P To avoid conflicts, be sure not to schedule the Deployment Backup and Recovery tool to run at the same time as the stand-alone metadata server backups. By default, the stand-alone metadata server backups (described in "About the SAS Metadata Server Backup and Recovery Facility" on page 169) are scheduled to run at 1:00 a.m. server local time every day except Sunday.

Review the Backup Policy

Next, review the backup policy by viewing the Policy page in SAS Backup Manager or by using the command sas-display-backup-config.

Determine what changes need to be made to the policy. You might need to specify alert email addresses, specify the threshold for alert emails, modify the backup retention period, specify a central vault location, filter backup sources, or add custom directories.

Specifying Alert Email Addresses

Effective with SAS 9.4M1, an alert email is generated if a backup or recovery is unsuccessful. By default, the email is sent to the system administrator email address that was specified in the SAS Deployment Wizard. To specify different email addresses, you can use SAS Backup Manager or the command sas-update-backup-config.

Specifying the Alert Email Threshold

By default, alert emails are generated for both failures and warnings. If you want to limit alerts to failures only, follow these steps:

1. In the **Plug-ins** tab of SAS Management Console, navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **SAS Deployment Backup and Recovery** ⇨ **Properties**.

2. On the **Advanced** tab of the **Properties** dialog box, locate the property svrbakupevm.email.notification.level. Change the value for this property from **ANALYSIS_WARN** to **FAIL**.

3. Click **OK**.

Modifying the Backup Retention Period

By default, backup files are retained for a period of 30 days. You can use either SAS Backup Manager or the command sas-update-backup-config to specify a different retention period.

Filtering Backup Sources

Effective with SAS 9.4M3, you can define filters that specify tiers, servers, databases, directories, or files to include or exclude from backups. For the metadata server, the filter definition can contain an option to reorganize the metadata repositories.

For example, you might need to exclude one or more of the automatically included directories (or specific files or subdirectories that they contain) for a particular **Levn** instance. Or you might want to exclude a particular database that is managed by a SAS Web Infrastructure Platform Data Server.

To define a filter, use the command sas-update-backup-config. Be sure to refer to "Define a Backup Filter: Best Practices" on page 419.

Adding Custom Directories

By default, the Deployment Backup and Recovery tool backs up the contents of the **Data** directories, **SASEnvironment** directories, and server configuration directories for each server on the SAS server tier. (If symbolic links in these directories point to other locations, the referenced locations are not backed up.)

You can specify additional "custom" directories that are to be backed up. Each of the directories must be located under ***SAS-configuration-directory*/Levn** on a host machine that is included in the backup.

Effective with SAS 9.4M3, you can specify custom directories by using the command sas-update-backup-config. If your deployment is not current with the third maintenance release, then use the command sas-add-backup-customdir.

Specifying or Changing a Central Vault Location

About Central Vault Locations

By default, backup files are stored locally on the same machine where the backup source is located. All backup sources are backed up to the following path on their respective host machines: ***SAS-configuration-directory*/Lev1/Backup/Vault**. This directory is created on each machine the first time a backup is executed.

You can also specify a central, network-accessible vault location to store backups. If a central vault is specified, backups are automatically copied from the various host machines to the central vault following each backup operation. To determine whether a central vault location has been specified for your deployment, view the **Policy** page in SAS Backup Manager. You can also execute the command sas-display-backup-config and check the vaultLocation value in the command output.

Note: If your deployment contains SAS 9.4 or SAS 9.4M1, a central vault location is required if your middle-tier environment includes a clustered SAS Content Server. Starting from SAS 9.4M2, use of a central vault location is optional. However, a central vault location is highly recommended to avoid the loss of backup files in the event that a host machine fails.

Specifying a Central Vault Location in a Homogeneous Operating System Environment

In homogeneous environments, you can use the SAS Deployment Wizard to specify a central vault location during the installation and configuration process. A homogeneous environment is one in which all of the host machines that are included in the backup are in the same operating system family. For example, Solaris and HP-UX machines are both considered to be in the UNIX operating system family.

If you did not specify the central vault location during installation and configuration, you can specify the location later. Follow these steps to specify a central vault location (or to change an existing one) in a homogeneous environment:

1. Create a directory that will serve as the central vault location. The directory must be in a network location that is accessible, via the same network path, to each host machine that is included in the backup.

2. Add the central vault location to your configuration by using one of the following methods:

   • Enter the vault location on the **Policy** page of SAS Backup Manager.

   • Execute sas-update-backup-config with the vault location specified in the JSON file.

   • Use SAS Management Console, as follows:

      1. In the **Plug-ins** tab of SAS Management Console, navigate to **Application Management ⇨ Configuration Manager ⇨ SAS Application Infrastructure ⇨ SAS Deployment Backup and Recovery ⇨ Properties**.

      2. On the **Advanced** tab of the **Properties** dialog box, locate the property svrbakupevm.shared.vault.dir, and enter the fully qualified path to the vault location.

      3. Click **OK**.

      4. If your deployment is at SAS 9.4M2 or earlier, then you must update BackupServer.ini to put the central vault location into effect. Perform these steps on each host machine that is included in the backup:

         i. Navigate to ***SAS-configuration-directory*/Lev1/Backup**.

         ii. Open the file BackupServer.ini in a text editor, and add the following property to the file:

```
backupserver.shared.vault.dir=absolute-central-backup-path
```

         On Windows machines, specify a UNC path, as shown in this example:

```
backupserver.shared.vault.dir=\\\\hostmachine1\\SAS\\CentralBackupVault
```

         Here is an example of this property on a UNIX host machine:

```
backupserver.shared.vault.dir=/hostmachine1/SAS/CentralBackupVault
```

         Be sure to specify the same path on each host machine.

   Note: All of the host machines that are included in the backup must be able to access the vault location using the path that you specify. On Windows machines, specify a UNC path.

3. On Windows systems, make sure that the backup user has Read and Write access to the central vault location. For information about specifying the backup user on Windows, see "Specifying a Backup User to Access a Central Vault Location or a Clustered Metadata Server (Windows Only)" on page 397.

On UNIX, make sure the SAS Installer user for each server and middle-tier machine has Read and Write access to the central vault location. After granting permissions on UNIX, you must restart the SAS Deployment Agent.

If your deployment is current with SAS 9.4M2 or SAS 9.4M3, no additional steps are necessary.

CAUTION:
> **Immediately after creating the central vault, it is strongly recommended that you perform a backup with either SAS Backup Manager or the sas-backup command. You cannot recover using local backups after a central vault has been defined.** After running the backup, check the central vault and verify that the backup files were successfully written.

Specifying a Central Vault Location in a Heterogeneous Operating System Environment

If your deployment is current with SAS 9.4M1, SAS 9.4M2, or SAS 9.4M3, you can use the following manual configuration steps to specify a central vault location in a heterogeneous operating system environment. A heterogeneous environment is one in which the host machines that are included in the backup are in different operating system families (for example, both UNIX and Windows).

1. When you use the SAS Deployment Wizard to install and configure your deployment, you should not select the option to configure a central vault location.

2. Create a directory that will serve as the central vault location. The directory must be in a network location that is accessible to each host machine that is included in the backup.

3. Use SAS Management Console to specify a user for backing up your Windows host machines. See "Specifying a Backup User to Access a Central Vault Location or a Clustered Metadata Server (Windows Only)" on page 397. The user's account must have Read and Write access to the central vault location. In addition, the account must be recognized by all of the Windows host machines in the deployment.

4. On UNIX, make sure the SAS Installer user for each server and middle-tier machine has Read and Write access to the central vault location. After granting permissions on UNIX, you must restart the SAS Deployment Agent.

5. Use the sas-backup command to run an ad hoc backup. This step creates a BackupServer.ini file on each host machine that is included in the backup.

6. On each host machine that is included in the backup, follow these steps to put the central vault location into effect:

   a. Navigate to ***SAS-configuration-directory*/Lev1/Backup**.

   b. Open the file BackupServer.ini in a text editor, and add the following property to the file:

      ```
      backupserver.shared.vault.dir=absolute-central-backup-path
      ```

      Make sure that all of the specified paths point to the same physical location. On each machine, specify a path that is appropriate for the operating system. On Windows machines, specify a UNC path, as shown in this example:

      ```
      backupserver.shared.vault.dir=\\\\hostmachine1\\SAS\\CentralBackupVault
      ```

      Here is an example of this property on a UNIX host machine:

      ```
      backupserver.shared.vault.dir=/hostmachine1/SAS/CentralBackupVault
      ```

CAUTION:

Immediately after creating or modifying the central vault configuration, it is strongly recommended that you perform a backup with either SAS Backup Manager or the sas-backup command. You cannot recover using local backups after a central vault has been defined. After running the backup, check the central vault and verify that the backup files were successfully written.

Removing a Central Vault Location

Follow these steps to remove a central vault location:

1. Remove the central vault location from your configuration by using one of the following methods:

   • Remove the vault location on the **Policy** page of SAS Backup Manager. SAS Backup Manager is included in SAS 9.4M3 and later.

   • Edit the JSON file that is used as input to sas-update-backup-config to remove the vault location. The JSON file should contain only the input for the parameter that is being changed, which in this case is the **vaultLocation** parameter.

   ```
   {
   "vaultLocation" : ""
   }
   ```

   Execute sas-update-backup-config with the JSON file that you just updated.

   • Use SAS Management Console, as follows:

     1. In the **Plug-ins** tab of SAS Management Console, navigate to **Application Management ⇨ Configuration Manager ⇨ SAS Application Infrastructure ⇨ SAS Deployment Backup and Recovery ⇨ Properties**.

     2. On the **Advanced** tab of the **Properties** dialog box, locate the property svrbakupevm.shared.vault.dir. Click the lock to unlock the property. Remove the vault location path from the property. Then click the lock again to lock the property.

     3. Click **OK**.

     4. If your deployment is at SAS 9.4M2 or earlier, then you must update BackupServer.ini to remove the central vault location. Perform these steps on each host machine that is included in the backup:

        i. Navigate to ***SAS-configuration-directory*/Lev1/Backup**.

        ii. Open the file BackupServer.ini in a text editor, and remove the following property from the file:

        ```
        backupserver.shared.vault.dir=absolute-central-backup-path
        ```

2. Remove the backup user.

   a. In SAS Management Console, open the User Manager and display the properties for the SAS Administrator user.

   b. In the Properties dialog box, select the **Accounts** tab and do the following:

      • Select the account for the SASDeploymentBackupAuth domain, and then click **Delete**.

      • At the confirmation window, click **Yes**.

3. Restart the SAS Web Application Server.

CAUTION:

Immediately after removing the central vault, it is strongly recommended that you perform a backup with either SAS Backup Manager or the sas-backup command. Verify that the backup completes successfully

Other Deployment Backup Administration Tasks

Specifying a Backup User to Access a Central Vault Location or a Clustered Metadata Server (Windows Only)

When Is a Backup User Required?

In either of the following situations, you must specify a special user account for the Deployment Backup and Recovery tool to use in performing backups:

- You are specifying a central vault location in an environment that includes one or more Windows hosts.

- You are configuring a clustered metadata server in an environment that includes one or more Windows hosts.

What Is a Backup User?

The backup user is a special external account that is defined as a login for the SAS Administrator user. The account is used by the SAS Deployment Agent to execute Deployment Backup and Recovery tool commands on each tier that is included in the backup. The account must meet the following requirements:

- The account must be an external account with credentials to log on locally to each of the following machines:

  - each of the Windows host machines that is included in the backup

  - the host machine that contains the central vault, if one is defined

  - each host machine on which a metadata server node has been configured

  For example, the account could exist in a Windows domain in which all of the machines participate.

  Note: In the local security policy for each machine in the preceding list, make sure that the account has the user right `Allow log on locally`.

- The account must be granted Read and Write permission to the following folders:

  - *SAS-configuration-directory*`/Lev1` on each Windows host machine that is included in the backup

  - the central vault, if one is defined

  - the following folder on the metadata server machine (or on each node in a metadata server cluster): *SAS-configuration-directory*`/Lev1/SASMeta/MetadataServer`

- The account must be granted Read permission to the metadata server's backup location. The default location is *SAS-configuration-directory*`/Lev1/SASMeta/MetadataServer/Backups`, but a different location might have been specified. To determine the location, open SAS Management Console and navigate to **Environment Management** ⇨ **Metadata Manager** ⇨ **Metadata Utilities** ⇨

Server Backup. Then right-click **Server Backup** and select **Backup Configuration**.

- The account must be used only for backups. Do not use an account such as sassrv that has been set up for another purpose.

Specifying a Backup User during Installation

In the following situations, the SAS Deployment Wizard prompts you to specify a backup user during installation:

- If you specify a central vault location in the wizard in an all Windows environment. (You can specify a central vault in the wizard only if you have a homogeneous operating system environment.)

- If you specify a Windows service login account in the wizard to support metadata server clustering.

Specifying a Backup User Manually

If a backup user was not specified during installation, you can use SAS Management Console to specify the user manually. You must specify a backup user manually in the following situations:

- if you create a central vault location after installation in either an all Windows environment or a heterogeneous environment

- if you configure metadata server clustering after installation in either an all Windows environment or a heterogeneous environment

Follow these steps to specify the user manually. You can also use these steps to change a previously defined backup user.

1. In SAS Management Console, open the User Manager and display the properties for the SAS Administrator user.

2. In the Properties dialog box, select the **Accounts** tab and do the following:

   - If the tab lists an account for the SASDeploymentBackupAuth domain, then select the account and click **Edit**. In the Edit Login Properties dialog box, enter the user ID and password for the new backup user.

   - If the tab does not list an account for the SASDeploymentBackupAuth domain, follow these steps:

     1. Click **New**.

     2. In the New Login Properties dialog box, enter the user ID and password for the backup user.

     3. Select the authentication domain **SASDeploymentBackupAuth**.

        If **SASDeploymentBackupAuth** does not appear in the drop-down list, select **New**, and then enter `SASDeploymentBackupAuth` in the New Authentication Domain dialog box.

   Note: Make sure that the user meets all of the requirements specified in "What Is a Backup User?" on page 397.

See Also

Disabling the Backup User on UNIX (in Deployments Not Current with the First, Second, or Third Maintenance Release)

In UNIX deployments that are not current with SAS 9.4M1, SAS 9.4M2, or SAS 9.4M3, backups will not work properly if you specified a backup user when you ran the SAS Deployment Wizard. In this situation, you must disable the backup user by following these steps:

1. In SAS Management Console, open the User Manager and display the properties for the SAS Administrator user.

2. On the **Accounts** tab of the Properties dialog box, select the account for the SASDeploymentBackupAuth domain.

3. Click **Delete**, and then click **OK**.

4. Restart the SAS Deployment Agent service using the SAS Installer account.

Note: If these steps are not taken, it would be necessary to configure sudo to enable the SAS Deployment Agent to execute processes as another user.

Note: If your deployment is current with SAS 9.4M1, SAS 9.4M2, or SAS 9.4M3, then no action is needed.

Specifying Which Web Infrastructure Platform Databases to Back Up

By default, the Deployment Backup and Recovery tool backs up all of the databases that are managed by the SAS Web Infrastructure Platform Data Server. If your deployment includes multiple instances of this server, the backup includes the databases that are managed by all of the servers. If you want the backup to include only a subset of these databases, you can modify the backup configuration to specify which databases to include.

Note: If your deployment is current with SAS 9.4M3, you can perform this task using the command sas-update-backup-config.

If your deployment is not current with 9.4M3, follow these steps on each machine where an instance of the SAS Web Infrastructure Platform Data Server has been configured:

1. Navigate to *SAS-configuration-directory*/**Lev1/Backup**.

2. Open the file BackupServer.ini in a text editor, and add the following property to the file:

   ```
   pg.backup.list.db.names=server-instance@database-1,
   server-instance@database-2...server-instance@database-n
   ```

   Here is an example of this property:

   ```
   pg.backup.list.db.names=Web Infrastructure Platform Data Server 9.4@Administration,
   Web Infrastructure Platform Data Server 9.4@SharedServices
   ```

Enter the server name exactly as it appears in the **Server Manager** tree in SAS Management Console.

A typical installation has the following databases:

Administration
> contains configuration information for the modules that SAS develops to extend the features of SAS Environment Manager.

EVManager
> contains configuration and metric information for the machines and servers that SAS Environment Manager manages in your deployment.

SharedServices
> is used by the SAS web applications and middle-tier software to store data such as comments. Digital content that is stored with SAS Content Server is also stored in this database.

> *Note:* The server and database names are case sensitive.

> *Note:* If you are using a third-party vendor database for the SharedServices database, the Deployment Backup and Recovery tool cannot back it up.

Depending on which SAS solutions you have licensed, additional databases might be configured.

Adding Forward Proxy Settings to the Deployment Backup Configuration

If the host machines that are included in the backup use a forward proxy to communicate with the SAS Web Server, then you must add the proxy settings to the backup configuration. Follow these steps:

1. Perform the following steps on the primary middle-tier machine. Also perform these steps on each of the other host machines from which you want to launch deployment backup and recovery commands.

   a. Navigate to **SAS-installation-directory/ SASPlatformObjectFramework/9.4/SASDeploymentBackup**, and open the file BackupServer.ini in a text editor.

   b. If you are not using a secure connection, add the following properties to the file:

   ```
   JavaArgs_10=-Dproxy.type=http
   JavaArgs_11=-Dhttp.proxyHost=host-name
   JavaArgs_12=-Dhttp.proxyPort=proxy-port
   ```

   If you are using a secure connection, add the following properties to the file:

   ```
   JavaArgs_10=-Dproxy.type=https
   JavaArgs_13=-Dhttps.proxyHost=host-name
   JavaArgs_14=-Dhttps.proxyPort=proxy-port
   ```

   In *host-name*, be sure to provide the fully qualified host name.

2. On each host machine from which you want to launch deployment backup and recovery commands, follow these steps:

   a. Navigate to **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin**.

   b. Open the file *command-name*.ini (for example, sas-backup.ini or sas-set-backup-schedule.ini) in a text editor.

   Add the same properties to the file that you added to BackupServer.ini. (See the preceding step 1.b).

   c. Repeat these steps for each Deployment Backup and Recovery command that you want to execute from this machine.

Deleting Files in Vault Directories after a Host Name Change

If this is the first time that the Deployment Backup and Recovery Tool has been run following a host name change, you should perform the following steps before using the tool:

1. Delete all the files that are present in the local vault directory on each tier. The path for the local vault directory on each tier is **SAS-configuration-directory/ Lev1/Backup/Vault**.

2. Delete all the files that are present in the shared vault directory.

3. Delete all the files that are present in the temporary directory **SAS-configuration-directory/Lev1/Web/WebAppServer/SASServer1_N/ temp/SASDeploymentBackup_APPHOME**.

   If SASServer1 is clustered then, do this for all SASServer1_N instances.

Overview of Backup and Recovery Commands

The following batch commands are available on each host machine to use in administering deployment backups and recoveries:

Note: Effective with SAS 9.4M5, the following commands are no longer available:

- sas-add-backup-customdir
- sas-display-backup-customdir
- sas-remove-backup-customdir
- sas-display-backup-source-config

Note: Effective with SAS 9.4M3, many of these functions can also be performed using the SAS Backup Manager user interface. SAS Backup Manager can be accessed from the **Administration** tab of SAS Environment Manager. For more information, see the *SAS Environment Manager 2.5 Administration: User's Guide*, *SAS Environment Manager 2.6 Administration: Fundamentals*, and the Help for SAS Backup Manager.

Command	Purpose	Documentation Location
sas-backup	Execute an ad hoc (unscheduled) backup.	"Perform a SAS Deployment Backup (sas-backup)" on page 406
sas-status-backup	Display the status of a backup.	"Display the Status of a Deployment Backup (sas-status-backup)" on page 409
sas-list-backups	Display detailed backup and recovery history.	"List Deployment Backups and Recoveries (sas-list-backups)" on page 410
sas-display-backup	Display information about a particular backup.	"Display a Deployment Backup (sas-display-backup)" on page 411

Command	Purpose	Documentation Location
sas-display-backup-source-content	Display detailed information about the contents of a specific backup that was taken from a particular source on a particular host machine.	"Display Backup Contents by Source Type and Host Name (sas-display-backup-source-content)" on page 411
sas-set-backup-schedule	Specify a day and time that is to be added to the backup schedule.	"Set the Deployment Backup Schedule (sas-set-backup-schedule)" on page 412
sas-list-backup-schedule	Display the current backup schedule.	"List the Deployment Backup Schedule (sas-list-backup-schedule)" on page 413
sas-remove-backup-schedule	Remove a specified day and time from the backup schedule.	"Remove Scheduled Deployment Backups (sas-remove-backup-schedule)" on page 414
sas-display-backup-config	Display the current backup configuration.	"Display the Deployment Backup Configuration (sas-display-backup-config)" on page 414
sas-display-backup-source-config	List backup configuration information that is associated with a specific backup source. *Note:* This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. Use the sas-display-backup-config command to perform the equivalent operation.	"Display the Deployment Backup Configuration by Source Type (sas-display-backup-source-config)" on page 416
sas-update-backup-config	Update the backup configuration. You can change the central vault location, the retention period, the email addresses for generated alerts, and whether scheduled backups are to run. Effective with for SAS 9.4M3, you can also define filters that specify tiers, servers, databases, directories, or files to include or exclude from the backup. For the metadata server, the filter definition can contain an option to reorganize the metadata repositories.	"Update the Deployment Backup Configuration (sas-update-backup-config)" on page 417
sas-add-backup-customdir	Specify custom directories that are to be backed up in addition to the directories that are included by default. *Note:* This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. Use the sas-update-backup-config command to perform the equivalent operation.	"Add Custom Directories to the Deployment Backup (sas-add-backup-customdir)" on page 426

Command	Purpose	Documentation Location
sas-display-backup-customdir	Display a list of custom directories that have been added to the backup. *Note:* This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. Use the sas-display-backup-config command to perform the equivalent operation.	"Display Custom Directories for the Deployment Backup (sas-display-backup-customdir)" on page 428
sas-remove-backup-customdir	Discontinue backing up a custom directory that was previously added. *Note:* This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. Use the sas-update-backup-config command to perform the equivalent operation.	"Command Reference for sas-recover-offline " on page 433
sas-recover-offline	Perform a full or partial recovery when some of the resources in the deployment are unavailable or have been taken offline to prevent user activity.	"Command Reference for sas-recover-offline " on page 433
sas-recover	If your deployment is SAS 9.4M1 or later, use sas-recover-offline. The command sas-recover is supported only for SAS 9.4 TS1M0.	"Command Reference for sas-recover" on page 435
sas-display-recovery	Display details about a recovery.	"Display a Deployment Recovery (sas-display-recovery)" on page 429

In addition to the detailed reference documentation for each command, be sure to refer to the "General Instructions for Using the Deployment Backup and Recovery Batch Commands" on page 403 and "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

General Instructions for Using the Deployment Backup and Recovery Batch Commands

The following instructions apply to the Deployment Backup and Recovery batch commands:

- The SAS Deployment Agent must be installed and running on the host machine for each component that is to be included in the backup. All commands except sas-list-backup-schedule, sas-set-backup-schedule, and sas-remove-backup-schedule require the agent to be running on all of these host machines. All other commands will fail if any of the hosts has a non-running agent.

  Host machines on which the SAS Deployment Agent is not installed are automatically excluded from backups, effective with SAS 9.4M2.

Note: If the agent has not been installed and configured, you can use SAS
Deployment Manager to install and configure it. See the "SAS Deployment
Agent" chapter of the *SAS Deployment Wizard and SAS Deployment Manager:
User's Guide*. For information about starting and stopping the agent, see Chapter
6, "Operating Your Servers," on page 61.

- On Windows 7, Windows 8, Windows Server 2008, and Windows Server 2012
  machines, you might need to open the command window with the Run as
  Administrator option to obtain the required level of access to directories and files. To
  enable this option, select **Start ⇨ All Programs ⇨ Accessories**. Then right-click
  Command Prompt and select **Run as administrator**.

- To locate the commands, go to the following path:

  **SAS-installation-directory/SASPlatformObjectFramework/9.4/
  tools/admin**

  Note: If your deployment is not current with SAS 9.4M2 or SAS 9.4M3, then you
  must run the commands from this directory.

- If your middle tier has been configured to use Transport Layer Security (TSL), be
  sure to specify **HTTPS** in the -protocol option. For information about this and other
  common options, see "Common Options for the Deployment Backup and Recovery
  Batch Commands" on page 405.

- If the command syntax contains an -inputdata parameter, you must supply the data in
  a file using the JavaScript Object Notation (JSON) format. Sample JSON files are
  provided in **SAS-installation-directory/
  SASPlatformObjectFramework/9.4/tools/admin/conf/sample**.

  Note: You cannot place the input data directly on the command line.

- After executing a command, you can view generated log messages in the following
  log files:

  - The following log file is created on the middle-tier machine:

    **SAS-configuration-directory/Lev1/Web/Logs/SASServer1_1/
    SASDeploymentBackup9.4.log**

    `TIP` As a best practice, you should check this log file (or run sas-list-backups)
    regularly to ensure that your backups are running successfully.

    By default, this log reports only errors and warnings. If you want to set different
    logging levels, you can do so by editing SASDeploymentBackup-log4j2.xml,
    which is located in **SAS-configuration-directory/Lev1/Web/
    Common/LogConfig/**.

    Note: For releases prior to SAS 9.4M8, the file that you edit is
    SASDeploymentBackup-log4j.xml.

  - The REST utility log is created in the directory from which the command was
    executed. For example:

    **SAS-installation-directory/
    SASPlatformObjectFramework/9.4/tools/admin/
    backuprestbatch.log**

  - For backup, recovery, and purge operations, log files are created in the directories
    where local backups are stored. The default location is as follows:

    SAS-configuration-directory/Lev1/Backup/Logs/*backup-ID*

  - The following log file might be useful to obtain information about server-side
    activity:

`SAS-configuration-directory/Lev1/Backup/backupserver.log`

Common Options for the Deployment Backup and Recovery Batch Commands

When submitting a deployment backup or recovery command (except sas-recover-offline), you must provide connection options to log on to the SAS Web Server or SAS Web Application Server. These options, which are represented in syntax statements as *web-connection-options*, are as follows:

Table 28.1 *Web Connection Options for Deployment Backup and Recovery Batch Commands*

Option	Description
-host *host-name*	Identifies the host machine for the SAS Web Server. If your deployment does not include SAS Web Server, specify the host machine for the SAS Web Application Server.
	This option is required if the -profile option is not set.
	Note: You must provide a fully qualified host name (for example, host1.mycompany.com). Do not specify *localhost*.
-port *port*	Specifies the port on which the SAS Web Server runs. If your deployment does not include the SAS Web Server, specify the port on which the SAS Web Application Server runs.
	This option is required if the -profile option is not set.
-user *user-ID*	Specifies the user ID of the SAS Administrator user (typically sasadm@saspw). This option is required if the -profile option is not set.
-password *password*	Specifies the password of the specified user. The password should be encrypted using SAS proprietary 32-bit encryption. To obtain the encrypted password, use PROC PWENCODE.
	This option is required if the -profile option is not set.
-protocol HTTP | HTTPS	Specifies the communications protocol that is used by the specified host machine and port. If this option is not specified, the default protocol (HTTP) is assumed.
	You can specify this option either on the command line or in the file that is specified in the -profile option.

Option	Description
-profile *filename*	Specifies the name of a file that contains the host, port, user ID, and password options. It can also contain the -protocol option if necessary. This option can be provided in place of -host, -port, -user, -password, and -protocol.
	Note: A sample profile file called environment.properties is located in **SAS-installation-directory/SASPlatformObjectFramework/9.4/tools/admin/conf/sample**.
	Note: If you use this file, be sure to use operating system controls to protect access to the file.
	Note: The password should be encrypted using SAS proprietary 32-bit encryption. To obtain the encrypted password, use PROC PWENCODE.
	Note: Make sure that the file specifies the fully qualified host name (for example, host1.mycompany.com). Do not specify *localhost*.

Note: The sas-recover-offline command uses different connection options. See "Command Reference for sas-recover-offline " on page 433.

The following additional options can be specified with any of the deployment backup batch commands:

Table 28.2   Other Common Options for the Deployment Backup and Recovery Batch Commands

Option	Description
-maxattempt *maximum-number-of-attempts*	The maximum number of attempts that are to be made to execute the command if the first attempt fails. The default value is 2.
-help	Displays online help for the command.

Deployment Backup Command Reference

Perform a SAS Deployment Backup (sas-backup)

Use sas-backup to execute an ad hoc (unscheduled) deployment backup using the current configuration settings. You can launch the command from any host machine that is included in the backup, and this single command backs up content from all of the participating host machines.

TIP   You can also use the SAS Backup Manager user interface to perform a backup.

The command has the following syntax:

sas-backup *web-connection-options* <-async> <-inputdata *input-data-pathname*>

The following list provides details:

web-connection-options
> See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-async
> specifies that the backup is to execute asynchronously. If you specify this option, the backup runs in background and displays a message stating that the backup is running. If you omit this option, the backup runs synchronously and displays a message when execution is completed.

-inputdata *input-data-pathname*
> specifies the path and filename of a JSON file that contains input data for the command.
>
> *Note:* For commands with the -inputdata parameter, you must supply the input data in a file. Do not place the input data directly on the command line.
>
> The input data is not required for this command. The file specifications are as follows. These specifications are effective with SAS 9.4M1.

Labels and Values	Description
"name": "*backup-name*"	Specifies an optional name for the backup. The name must be unique. It is recorded in backup history and is displayed with backup status information.
"comment": "*comment*"	Specifies an optional free-form comment describing the backup. The comment is recorded in backup history and is displayed with backup status information.

> Here is an example input file:
>
> ```
> {
>     "name": "My Backup",
>     "comment": "Unscheduled backup"
> }
> ```
>
> A sample input file is provided at **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin/conf/sample/sas- backup.json**.
>
> Here is an example of the displayed output:

```
sas-backup
    id:     2013-11-15T03_44_32-0500
    name:   firstbackup
    type:   backup
    status: completed
    size:   313841192
    comment:            testingsync
    scheduled:          false
    runBy: sasadm
    startTimeStamp: 2013-11-15T08:44:38Z
    endTimeStamp:   2013-11-15T08:47:05Z
    offlineBackup:  false
    backupSources
    sourceType:     metadataServer
    status: completed
```

```
    size:    145916864
    hosts

      hostName:      myhost.company.com
      status: completed
      size:    145916864
      backUpLocation: C:\SAS\Config\Lev1\SASMeta\MetadataServer\Backups\2013-11-15T03_44_38-05_00
      osType: wx6
      sasHome:       C:\Program Files\SASHome
      configDirectory:      C:\SAS\Config\Lev1
backupSources
  sourceType:    contentServer
  status: completed
  size:    127013301
  hosts
    hostName:      myhost.company.com
    status: completed
    size:    127013301
    logFilePath:    C:\SAS\Config\Lev1\Backup\Logs\2013-11-15T03_44_32-0500\ContentServer
    backUpLocation: C:\SAS\Config\Lev1\Backup\Vault\2013-11-15T03_44_32-0500\content
    osType: wx6
    sasHome:       C:\Program Files\SASHome
    configDirectory:      C:\SAS\Config\Lev1
backupSources
  sourceType:    configDirectories
  status: completed
  size:    38092508
  hosts
    hostName:      myhost.company.com
    status: completed
    size:    38092508
    logFilePath:    C:\SAS\Config\Lev1\Backup\Logs\2013-11-15T03_44_32-0500\ConfigDir
    backUpLocation: C:\SAS\Config\Lev1\Backup\Vault\2013-11-15T03_44_32-0500\levconfig
    osType: wx6
    sasHome:       C:\Program Files\SASHome
    configDirectory:      C:\SAS\Config\Lev1
backupSources
  sourceType:    customDirectories
  status: completed
  size:    1022
  hosts
    hostName:      myhost.company.com
    status: completed
    size:    1022
    logFilePath:    C:\SAS\Config\Lev1\Backup\Logs\2013-11-15T03_44_32-0500\UserDefinedFolders
    backUpLocation: C:\SAS\Config\Lev1\Backup\Vault\2013-11-15T03_44_32-0500\customdata
    osType: wx6
    sasHome:       C:\Program Files\SASHome
    configDirectory:      C:\SAS\Config\Lev1
backupSources
  sourceType:    database
  status: completed
  size:    2817497
  hosts
    hostName:      myhost.company.com
    status: completed
```

```
    size:   2817497
    logFilePath:    C:\SAS\Config\Lev1\Backup\Logs\2013-11-15T03_44_32-0500\Database
    backUpLocation: C:\SAS\Config\Lev1\Backup\Vault\2013-11-15T03_44_32-0500\db
    osType:   wx6
    sasHome:        C:\Program Files\SASHome
    configDirectory:        C:\SAS\Config\Lev1
    sourceInstances
        instanceName: Web Infrastructure Platform Data Server 9.4
        host:   myhost.company.com
        port:   9432
        sourceEntries
                name:   postgres
                size:   1892
                status: completed
                status: completed
                name:   Administration
                size:   9251
                status: completed
                name:   SharedServices
                size:   502300
                status: completed
                name:   EVManager
                size:   2292089
                status: completed
        sourceInstances
  hosts
 backupSources
```

> *Note:* The **`backUpLocation`** field contains the location of the backup files on the local machine.

Display the Status of a Deployment Backup (sas-status-backup)

Use sas-status-backup to display status information for a particular backup. You can use this command to display the status of a backup operation that is running in asynchronous mode.

> **T I P** The SAS Backup Manager user interface also displays the status of backups and recoveries.

The command has the following syntax. This syntax is effective with SAS 9.4M1.

sas-status-backup *web-connection-options* -backupid *backup-ID*

The following list provides details:

web-connection-options
See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-backupid *backup-ID*
specifies the identifier of the backup, which is also the name of the backup directory (for example, **`2013-02-01T03_13_01`**).

The displayed output has the same format as the output from the sas-backup command.

List Deployment Backups and Recoveries (sas-list-backups)

Use sas-list-backups to display information about backups and recoveries that are recorded in backup history, including backups that have been purged due to the retention policy.

TIP The SAS Backup Manager user interface also displays a list of backups and recoveries.

The command has the following syntax. This syntax is effective with SAS 9.4M1.

sas-list-backups

> *web-connection-options* <-operationtype backups | restores>

The following list provides details:

web-connection-options
> See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-operationtype backups | restores
> specifies whether backups or restores are to be listed. If this parameter is omitted, both backups and restores are listed.
>
> *Note:* This parameter must be specified using lowercase letters.

Here is an example of the displayed output:

```
=========================== Backup History ===========================
sas-list-backups
        name:   BackupHistory
                items
                id:     2017-05-23T06_47_40
                type:   Backup
                status: completed
                size:   40568417
                runBy:  sasadm
                scheduled:      false
                startTimeStamp: 2017-05-23T10:47:51.000Z
                endTimeStamp:   2017-05-23T10:48:58.000Z
                purged: false
                backupSources
                        sourceType:     metadataServer
                        status: completed
                        size:   21568211

                sourceType:     contentServer
                        status: completed
                        size:   7253568

                sourceType:     configDirectories
                        status: completed
                        size:   102543

                sourceType:     database
                        status: completed
                        size:   11644095

=========================== Recovery History ===========================
```

```
sas-list-backups
        name:   RecoveryHistory
            items
            id:     2017-05-23T06_50_40
            type:   Recovery
            status: completed
            runBy:  sasadm
            scheduled:      false
            startTimeStamp: 2017-05-23T10:50:52.000Z
            endTimeStamp:   2017-05-23T10:53:09.000Z
            backupSources
                    sourceType:     metadataServer
                    status: completed
                    size:   21413888

        sourceType:     contentServer
                status: completed
                size:   7253568

        sourceType:     configDirectories
                status: completed
                size:   102543

        sourceType:     database
                status: completed
                size:   11644095
```

Display a Deployment Backup (sas-display-backup)

Use sas-display-backup to display information about a particular backup that is recorded in backup history.

TIP The SAS Backup Manager user interface also displays details about backups.

The command has the following syntax:

sas-display-backup *web-connection-options* -backupid *backup-ID*

The following list provides details:

web-connection-options
See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-backupid *backup-ID*
specifies the identifier of the backup, which is also the name of backup directory (for example, **2013-02-01T03_13_01**).

The displayed output has the same format as the output from the sas-backup command.

Effective with SAS 9.4M3, **purged: True** is displayed (instead of **offlineBackup: True**) if backup files have been purged due to the retention policy.

Display Backup Contents by Source Type and Host Name (sas-display-backup-source-content)

Use sas-display-backup-source-content to display detailed information about the contents of a specific backup that was taken from a particular source on a particular host machine. The command has the following syntax:

sas-display-backup-source-content

> *web-connection-options*
>
> -backupid *backp-id*
>
> -sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE
> | CUSTOMDIRECTORIES | METADATASERVER
>
> -hostname *host-name*

The following list provides details:

web-connection-options
> See "Common Options for the Deployment Backup and Recovery Batch
> Commands" on page 405.

-backupid *backup-ID*
> specifies the identifier of the backup, which is also the name of backup directory (for
> example, `2013-02-01T03_13_01`).

**-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE |
  CUSTOMDIRECTORIES**
> specifies the backup source for which content details are to be displayed. The source
> names are not case sensitive.
>
> *Note:* METADATASERVER is not a valid source type for this command. For the
>   contents of the metadata server backup, look in the backup location for the
>   metadata server backup utility (for example, `SAS-configuration-`
>   `directory/Lev1/SASMeta/MetadataServer/Backups`).
>
> *Note:* If a central vault location has been defined, then you cannot specify
>   CONFIGDIRECTORIES or CUSTOMDIRECTORIES.

-hostname *host-name*
> specifies the fully qualified host name of the machine whose backup content details
> are to be displayed.

Here is a partial example of the displayed output for the CONTENTSERVER source
type:

```
sas-display-backup-source-content
    directoryContents
        directoryPath:  /repository/datastore/74/7a/dc
        files
            fileName:     747adc684012b15842e726b06dd0a34c6e8f9e52.MD5
            fileName:     747adc684012b15842e726b06dd0a34c6e8f9e52
        directoryPath:  /repository/datastore/f2/09/50
        files
            fileName:     f20950151c7326f555e2ff33829e83f5bdaca59c.MD5
            fileName:     f20950151c7326f555e2ff33829e83f5bdaca59c
```

Set the Deployment Backup Schedule (sas-set-backup-schedule)

Use sas-set-backup-schedule to specify a day and time that is to be added to the
deployment backup schedule.

> *TIP*   You can also use the SAS Backup Manager user interface to modify the backup
> schedule.

> *TIP*   To avoid conflicts, be sure not to schedule the deployment backup at the same
> time as the stand-alone metadata server backups. By default, the stand-alone
> metadata server backups (described in "About the SAS Metadata Server Backup and

Recovery Facility" on page 169) are scheduled to run at 1:00 a.m. server local time every day except Sunday.

The command has the following syntax:

sas-set-backup-schedule *web-connection-options* -inputdata *input-data-pathname*

The following list provides details:

web-connection-options
   See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-inputdata *input-data-pathname*
   specifies the path and filename of a JSON file that contains input data for the command. The file specifications are as follows. These specifications are effective with SAS 9.4M1.

Labels and Values	Description
`"dayOfWeek": "monday"` | `"tuesday"` | `"wednesday"` | `"thursday"` | `"friday"` | `"saturday"` | `"sunday"`	Specifies a day of the week that the backup is to execute.
`"time": "`*hh:mm*`"`	Specifies the time, based on a 24-hour clock, that the backup is to execute.

Effective with SAS 9.4M1, each input file can contain only one day and time. If you want to add more than one day and time to the schedule, you must submit a separate command (with a different input file) for each day and time.

Here is an example input file:

```
{
    "dayOfWeek": "SUNDAY",
    "time": "13:45"
 }
```

A sample input file is provided at **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin/conf/sample/sas- set-backup-schedule.json**.

Note: Backup days and times that you specify in this command add to (rather than replace) days and times that you previously scheduled. If you want to remove specific days and times from the schedule, use sas-remove-backup-schedule.

Note: For scheduled backups to run, **"scheduledBackupsAllowed"** must be set to **"true"** (the default setting) in your backup configuration. See "Update the Deployment Backup Configuration (sas-update-backup-config)" on page 417.

List the Deployment Backup Schedule (sas-list-backup-schedule)

Use sas-list-backup-schedule to display the deployment backup schedule that is currently in effect.

TIP The SAS Backup Manager user interface also displays the backup schedule.

The command has the following syntax:

sas-list-backup-schedule *web-connection-options*

For *web-connection-options*, see "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

Here is an example of the displayed output:

```
name:    ScheduleList
start:   1
limit:   1
count:   1
items
        dayOfWeek:  Sunday
        time:   01:00
            id:     sun0100
```

Remove Scheduled Deployment Backups (sas-remove-backup-schedule)

Use sas-remove-backup-schedule to remove a specified day and time from your deployment backup schedule.

TIP You can also use the SAS Backup Manager user interface to remove a scheduled backup.

The command has the following syntax:

sas-remove-backup-schedule *web-connection-options* -id *backup-schedule-ID*

Note: This syntax is effective with SAS 9.4M2. Previously, the command required a JSON input file instead of a schedule ID.

The following list provides details:

web-connection-options
See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-id *backup-schedule-ID*
specifies the ID of the schedule entry that you want to delete. Schedule IDs are generated automatically using the format *dayhhmm* (for example, **sun0100**).

To determine the IDs of the schedule entries that are currently in effect, use sas-list-backup-schedule.

Note: If you want to temporarily discontinue scheduled backups without modifying the schedule information, you can set **"scheduledBackupsAllowed"** to **"false"** in your backup configuration. See "Update the Deployment Backup Configuration (sas-update-backup-config)" on page 417.

Display the Deployment Backup Configuration (sas-display-backup-config)

Use sas-display-backup-config to list the configuration properties that are currently in effect for your deployment backups.

TIP The SAS Backup Manager user interface also displays the backup configuration.

Effective with SAS 9.4M3, this command also creates a JSON file called sas-display-backup-config.json that describes the current backup configuration. The file is created in the directory from which you issue the command. The file is useful when you want to

add a filter to your backup configuration. See "Define a Backup Filter: Best Practices" on page 419.

The command has the following syntax:

sas-display-backup-config

> *web-connection-options*

> <-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE
>> | CUSTOMDIRECTORIES | METADATASERVER>

Note: This syntax is effective with SAS 9.4M3. In previous SAS 9.4 releases, the syntax includes only the web connection options.

The following list provides details:

web-connection-options
> See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE | CUSTOMDIRECTORIES | METADATASERVER
> specifies the backup source for which configuration information is to be displayed. The source names are not case sensitive.

Here is an example of the output from this command. This output format is effective with SAS 9.4M3:

```
name:   Configurations
items
  id:     default
  vaultLocation:
  emailToList
    user1@example.com
    scheduledBackupsAllowed:        true
    daysToRetainBackup:     30
    sourceconfiguration
      sourceType:     metadataServer
      configurableServers
        name:   SASMeta - Logical Metadata Server
        sasTierId:      tier1
        included:       true

      sourceType:     contentServer
      configurableServers
        name:   SAS Content Server
        sasTierId:      tier1
        included:       true

      sourceType:     configDirectories
      configurableServers
        id:     tier1
        name:   tier1
        sasTierId:      tier1
        configurablePath:       C:\SAS\Config\Lev1
        included:       true
      filter
        includes
          C:\SAS\Config\Lev1\SASApp/Data,C:\SAS\Config\Lev1\SASApp/SASEnvironment
```

```
            sourceType:     database
            configurableServers
              name:   Web Infrastructure Platform Data Server9.4
              sasTierId:      tier1
              included:       true


    tierConfiguration
      sasTiers
        id:     tier1
        tierName:       tier1
        sasHome:        C:\Program Files\SASHome
        sasConfig:      C:\SAS\Config\Lev1
        included:       true
      tierHost
        hostType:       wx6
        hostName:       myhost.company.com


The json output is written to the file: C:\Program Files\SASHome\SASPlatformObje
ctFramework\9.4\tools\admin\sas-display-backup-config.json
```

Display the Deployment Backup Configuration by Source Type (sas-display-backup-source-config)

> *Note:* This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. For equivalent functionality, see sas-display-backup-config on page 414.

Use sas-display-backup-source-config to list figuration information that is associated with a specific backup source. The command has the following syntax:

sas-display-backup-source-config

> *web-connection-options*
>
> -sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE
> | CUSTOMDIRECTORIES | METADATASERVER

The following list provides details:

web-connection-options
    See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE | CUSTOMDIRECTORIES | METADATASERVER
    specifies the backup source for which configuration information is to be displayed. The source names are not case sensitive.

Here is an example of the displayed output:

```
sas-display-backup-source-config
    sourceType:     configDirectories
    hosts
        hostName:   myhost.company.com
        osType:     wx6
        sasHome:    C:\Program Files\SASHome
        configDirectory:    C:\SAS\Config\Lev1
```

Update the Deployment Backup Configuration (sas-update-backup-config)

About sas-update-backup-config

Use sas-update-backup-config to update the backup configuration properties that are in effect for the deployment. You can do the following:

- Update your site's basic backup configuration, including the number of days to retain backups, email addresses for alerts, and whether scheduled backups are to run. In deployments with homogeneous operating system environments, you can also use this command to specify or update the central vault location.

- Define filters that specify tiers, servers, databases, directories, or files to include or exclude from the backup. For the metadata server, the filter definition can contain an option to reorganize the metadata repositories.

The configuration changes and filters affect all subsequent scheduled and ad hoc backups.

> **TIP** You can also use the SAS Backup Manager user interface to update the basic backup configuration. You cannot use the user interface to define filters.

Syntax for sas-update-backup-config

The sas-update-backup-config command has the following syntax.

sas-update-backup-config *web-connection-options*

    <-inputdata *input-data-pathname*>

    <-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE
        | CUSTOMDIRECTORIES | METADATASERVER>

    <-id *tier-id*>

    <-d>

Note: This syntax is effective with SAS 9.4M3. In previous SAS 9.4 releases, the syntax includes only the web connection options and the -inputdata option.

Syntax Description for sas-update-backup-config

web-connection-options
    See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-inputdata *input-data-pathname*
    specifies the path and filename of a JSON file that contains input data for the command. Omit this option if you are specifying the -d option to delete a filter.

    For file specifications, see the topic for the type of update you are making:

- "Define the Basic Backup Configuration" on page 418
- "Define a Filter to Include or Exclude Specific Tiers" on page 420
- "Define a Server or Database Filter" on page 421
- "Define a Filter for Configuration Directories and Custom Directories" on page 423

-sourcetype CONFIGDIRECTORIES | CONTENTSERVER | DATABASE | CUSTOMDIRECTORIES | METADATASERVER | SASTIERS
specifies the backup source type for which a filter is being defined, modified, or deleted. The source names are not case sensitive. This option is new with SAS 9.4M3.

-id *filter-id*
identifies an existing filter that is being modified or deleted. You can specify only one ID. This option is new with SAS 9.4M3.

-d
deletes the filter that is identified in -id. This option is new with SAS 9.4M3.

Note: You cannot add and delete filters for SASTIERS and CONFIGDIRECTORIES source types. Instead, modify the filter.

The following topics provide JSON file specifications, as well as example JSON files and commands, for each type of configuration update.

Define the Basic Backup Configuration

The file specifications for updating the basic backup configuration are as follows. These specifications are effective with SAS 9.4M1.

Table 28.3   Input Data for Basic Backup Configuration

Labels and Values	Description
`"vaultLocation"`: "*backup-path*"	Specifies the path to the central vault location. All of the host machines that are included in the backup must be able to access the vault location using this path. On Windows machines, specify a UNC path.
	Note: You can use this parameter to specify or change the central vault location only if you have a homogeneous operating system environment. In addition, you must perform the manual steps that are documented in "Specifying a Central Vault Location in a Homogeneous Operating System Environment" on page 394.
	Note: In heterogeneous operating system environments, you must omit this parameter. If you need to specify or update the central vault location, use the manual steps that are documented in "Specifying a Central Vault Location in a Heterogeneous Operating System Environment" on page 395.
`"daysToRetainBackup"`: *retention-days*	Specifies the number of days that backup files are to be retained in the central vault location and in the local backup location on each machine. Backups directories and their contents will be deleted automatically after this number of days.
	If no retention period is specified, backups are retained for 30 days by default.
`"scheduledBackupsAllowed"`: `"true"` | `"false"`	Specifies `"true"` or `"false"` to indicate whether scheduled backups are to run.
	The default setting is `"true".`

Labels and Values	Description
"emailToList": ["_email-address-1_**","**_email-address-2_**",...]**	Specifies one or more email addresses to which alert messages are to be sent when a backup or a recovery is unsuccessful. This parameter is new with SAS 9.4M1.
	If you do not specify this option, alert messages are sent to the system administrator email address that was specified in the SAS Deployment Wizard.
	If you do not want alert emails to be generated, specify a blank email address as follows: **"emailToList":[""]**
	By default, alert emails are generated for both failures and warnings. Effective with the third maintenance release of SAS 9.4, you can limit alert emails to failures only. See "Specifying the Alert Email Threshold" on page 392.

Here is an example input file for updating the basic backup configuration:

```
{
        "vaultLocation":"\\\\hostmachine1\\SAS\\CentralBackupVault",
        "daysToRetainBackup": 45,
        "scheduledBackupsAllowed": "true",
        "emailToList":["Bill@mycompany.com","Susan@mycompany.com"]
}
```

Here is an example command for updating the basic backup configuration:

```
sas-update-backup-config -profile environnment.properties
-inputdata conf/sample/sas-update-backup-config.json
```

If you are specifying or changing the central vault location, be sure to also perform the manual steps that are documented in "Specifying or Changing a Central Vault Location" on page 393.

Define a Backup Filter: Best Practices

Note: The filter capability is new with SAS 9.4M3.

For best results, follow these steps when using sas-update-backup-config to define a backup filter. You can use a backup filter to specify tiers, servers, databases, directories, or files to include or exclude from the backup.

1. First run the sas-display-backup-config command. This command creates a JSON file called sas-display-backup-config.json that describes the current backup configuration. The file is created in the directory from which you issue the sas-display-backup-config command.

2. Open sas-display-backup-config.json, and locate the statements containing the names-and-value pairs that pertain to the tier, server, directories, or files that you want to filter.

3. Copy the statements and paste them into a new JSON file. Then edit the statements as needed to specify the filter. Be sure to do the following:

    •   remove any name-value-pairs that have a value of null

    •   change the "included" value to true or false as needed

    •   insert an opening bracket ({) at the beginning of the file and a closing bracket (}) at the end of the file

4. Run sas-update-backup-config with the new JSON file as input. Be sure to specify the correct -sourcetype option.

5. After running sas-update-backup-config, verify the configuration by doing one of the following:

   - Run sas-display-backup-config.

   - View the source details on the **Policy** page in SAS Backup Manager.

   - Run an ad hoc backup to test the filter.

6. The filter affects all subsequent scheduled and ad hoc backups. When a filter is no longer needed, take one of the following actions:

   - Modify the filter. See "Modify a Filter" on page 425.

   - Delete the filter. See "Delete an Existing Filter" on page 426.

The following topics provide JSON file specifications, as well as example JSON files and commands, for each type of filter.

Define a Filter to Include or Exclude Specific Tiers

Note: This capability is new with SAS 9.4M3.

The Deployment Backup and Recovery tool automatically discovers all instances of **SAS-configuration-directory/Levn** directories. The tool identifies each of these instances as a separate tier and automatically includes each of the discovered tiers in the backup.

Situations might occur in which you need to exclude one or more of the discovered tiers. For example, the backup tool might discover a **SAS-configuration-directory/Levn** directory that is on a client machine rather than a server or middle-tier machine.

Note: It is strongly recommended that you not exclude the SAS Metadata Server tier from backups.

To exclude one or more tiers, include the following labels and values in the JSON file:

Table 28.4   *Input Data for Filtering Tiers*

Labels and Values	Description
"tierName": "*tier-name*"	Specifies the name of a tier that is to be included or excluded from backups. You can obtain the tier name from sas-display-backup-config.json. It is also displayed in the tierConfiguration section of the output from sas-display-backup-config.
"included": true | **false**	Specifies **true** or **false** to indicate whether the specified tier is to be included in backups. The value must be specified using lowercase letters. The default setting is **true.**

For example, the following JSON file excludes **tier4** from all subsequent backups:

```
{
 "tierName":"tier4",
 "included": false
}
```

Here is an example command for filtering tiers:

```
sas-update-backup-config -profile environnment.properties -sourcetype sastiers
-inputdata conf/sample/sas-update-backup-config.json -id tier4
```

Define a Server or Database Filter

Note: This capability is new with SAS 9.4M3.

SAS recommends backing up all instances of the SAS Metadata Server, SAS Content Server, and SAS Web Infrastructure Platform Data Server. In addition, SAS recommends backing up all databases that are managed by SAS Web Infrastructure Platform Data Servers. All of these servers and databases are included in backups by default. However, situations might occur in which you need to exclude a server instance (or one or more databases) from a backup.

In addition, for the metadata server, you might want to reorganize the metadata repositories to reclaim unused disk space.

To define a server or database filter, include the following labels and values in the JSON file:

Table 28.5   *Input Data for Filtering Server Instances*

Labels and Values	Description
"name": "*database-server-name*"	Specifies the name of a particular server instance, exactly as it appears in the JSON file that is created by sas-display-backup-config (for example, **SASMeta - Logical Metadata Server**, **SAS Content Server**, or **Web Infrastructure Platform Data Server 9.4**). *Note:* You can specify only one server instance in each JSON file. If you need to filter multiple server instances, run the command separately for each.
"sasTierId": "*tier-ID*"	Specifies the tier ID of the tier where the server is deployed. To obtain the tier ID, run sas-display-backup-config and refer to the JSON file that it creates.
"included": **true** | **false**	Specifies **true** or **false** to indicate whether the specified server instance is to be included in backups. The value must be specified using lowercase letters. The default setting is **true.**
"filter":	(Optional) Introduces the list of the server's databases that are to be included or excluded. *Note:* This option applies only to instances of the SAS Web Infrastructure Platform Data Server.
"includes": "*database-name-1*","*database-name-n*"	(Optional) Specifies the SAS Web Infrastructure Platform Data Server's databases that are to be included in the backup (for example, **Administration**, **EVManager**, or **SharedServices**). All other databases are excluded. *Note:* Database names are case sensitive. *Note:* If a database is specified in both the **"excludes":** list and the **"includes":** list, then it is excluded.
"excludes": "*database-name-1*","*database-name-n*"	(Optional) Specifies the SAS Web Infrastructure Platform Data Server's databases that are to be excluded from the backup.
"useroptions":	(Optional) Introduces user options that are available for specific server types. *Note:* In the current release, a user option is available only for the SAS Metadata Server.
"options":	(Optional) Introduces a set of options under the **useroptions** label.

Labels and Values	Description
`"reorganizeMetadataRepositories":true|false`	(Optional) Specifies whether the metadata server's metadata repositories are to be reorganized during subsequent backups. This option reclaims unused disk space left from previously deleted metadata objects. *Note:* This option applies only to instances of the SAS Metadata Server. *Note:* The default schedule for the metadata server's backup facility specifies a weekly reorganization at 1:00 a.m. each Monday. It is not necessary to reorganize the repositories more frequently than once a week, except in extraordinary situations such as the deletion of a large amount of metadata.

Example: Filter to Exclude the SAS Content Server

The following example JSON file excludes the SAS Content Server from all subsequent backups:

```
{
"name" : "SAS Content Server",
        "sasTierId" : "tier1",
         "included" : false


}
```

Here is an example command for excluding SAS Content Server:

```
sas-update-backup-config -profile environnment.properties -sourcetype contentserver
-inputdata conf/sample/sas-update-backup-config.json
```

Example: Filter to Exclude the SharedServices Database

The following example JSON file excludes the SharedServices database from all subsequent backups of Web Infrastructure Platform Data Server 9.4:

```
{
  "name": "Web Infrastructure Platform Data Server 9.4",
  "sasTierId": "tier1",
  "included": true,
  "filter": {
    "excludes": [
      "SharedServices"
    ]
  }
}
```

Here is an example command for excluding a database:

```
sas-update-backup-config -profile environnment.properties -sourcetype database
-inputdata conf/sample/sas-update-backup-config.json
```

Example: Filter to Reorganize Metadata Repositories

The following example JSON file specifies that the metadata server's repositories are to be reorganized in subsequent backups:

```
{
  "name": "SASMeta - Logical Metadata Server",
  "sasTierId": "tier1",
  "included": true,
  "useroptions": {
```

```
      "options": {
        "reorganizeMetadataRepositories": "true"
      }
    }
  }
```

Here is an example command for setting the option to reorganize metadata repositories:

```
sas-update-backup-config -profile environnment.properties -sourcetype metadataserver
-inputdata conf/sample/sas-update-backup-config.json
```

Define a Filter for Configuration Directories and Custom Directories

Note: This capability is new with SAS 9.4M3.

Within each discovered instance of **SAS-configuration-directory/Levn**, SAS recommends backing up the contents of the **Data**, **SASEnvironment**, and server configuration directories. All contents of these directories are included by default. However, the following situations might occur:

- You need to exclude one or more of the automatically included directories (or specific files or subdirectories that they contain) for a particular **Levn** instance.

- You want to specify an additional directory (referred to as a custom directory) that is to be backed up within a particular **Levn** instance. You can also define filters that specify which subdirectories and files are to be included or excluded from these backups.

To perform one of these tasks, include the following labels and values in the JSON file:

Table 28.6 *Input Data for Filtering Configuration Directories and Files*

Labels and Values	Description
"name": *"tier-name"*	Specifies the name of the tier where the filter is to be applied. You can obtain the tier name from sas-display-backup-config.json. It is also displayed in the tierConfiguration section of the output from sas-display-backup-config.
	If you are defining a custom directory, specify a new, unique name to identify the custom directory in the backup configuration. If you are modifying a previously defined custom directory, specify the existing name.
"sasTierId": *"tier-ID"*	Specifies the ID of the tier where the filter is to be applied. To obtain the ID, run sas-display-backup-config and refer to the JSON file that it creates. It is also displayed in the tierConfiguration section of the output from sas-display-backup-config.
"configurablePath": *"path"*	Specifies the full path to the **Levn** directory where the filter is to be applied (for example, **C:\\SAS\\Config\\Lev1** or **/install/cfgsas1/config/Lev1**).
	If you are defining a custom directory, specify the full path to the directory. It must be within **SAS-configuration-directory/Levn**.
"included": true | **false**	Specifies **true** or **false** to indicate whether the directory that is specified in configurablePath is to be included in backups.
	The default setting is **true.**
"filter":	Introduces an optional list of directories and files that are to be included or excluded.

Labels and Values	Description

"includes": *"directory-path-or-filename-1","directory-path-or-filename-2","directory-path-or-filename-n"*	Introduces a list of directories or files within the directory that is specified in **configurablePath** that are to be included in backups. You can specify one or more instances of the following:
	
	• the full path to a subdirectory within the configurable path
	
	• a filename, or a filename pattern using Ant regular expressions. For example:
	
	   • **"*.txt"** specifies all files with the suffix TXT within the specified directory.
	
	   • **"**/*.xml"** specifies all files with the suffix XML that are in the specified directory or its subdirectories
	
	   • **"**/Logs/**"** specifies all files that are in a Logs directory within the specified subdirectory, or in subdirectories of the Logs directory.
	
	Note: Directories and files that are backed up by default will continue to be backed up regardless of whether you specify them in the **"includes":** list. Use the **"excludes":** parameter if you want to exclude directories or files that are backed up by default.

"excludes": *"directory-path-or-filename","directory-path-or-filename"*	Introduces a list of directories or files within the directory that is specified in **configurablePath** that are to be excluded from backups. You can specify one or more instances of the following: • the full path to a subdirectory within the configurable path • a filename or a filename pattern (using Ant regular expressions). If the filter specifies subdirectories, then files are considered only if they are in included subdirectories. *Note:* If a subdirectory, filename, or filename pattern is specified in both the **"excludes":** list and the **"includes":** list, then it is excluded.

Example: Filter to Exclude Logs from Configuration Directory Backups

The following example JSON file excludes log files from backups of the server directories in **c:/sas/config/Lev1** of tier1.

```
{
  "name": "tier1",
  "sasTierId": "tier1",
  "configurablePath": "c:\\sas\\config\\Lev1",
  "included": true,
  "filter": {
      "includes": [
        "c:\\sas\\config\\Lev1\\SASApp\\Data",
        "c:\\sas\\config\\Lev1\\SASApp\\SASEnvironment"
      ],
      "excludes": [
        "**/*.log"
      ]
  }
}
```

Here is an example command for filtering configuration directories:

```
sas-update-backup-config -profile environnment.properties -sourcetype configdirectories
-inputdata conf/sample/sas-update-backup-config.json -id tier1
```

Example: Filter to Add a New Custom Directory

The following example JSON file specifies a new custom directory, **c:/sas/config/ Lev1/Utilities**, that is to be included in backups for tier1. Within this directory, only files with the suffix XML are to be included.

```
{
   "name": "customDir1",
   "configurablePath": "c:\\sas\\config\\Lev1\\Utilities",
   "sasTierId": "tier1",
   "included": true,
     "filter": {
         "includes": [
           "**/*.xml"
         ]
     }
}
```

Here is an example command for specifying a new custom directory:

```
sas-update-backup-config -profile environnment.properties -sourcetype customdirectories
-inputdata conf/sample/sas-update-backup-config.json
```

Modify a Filter

Note: This capability is new with SAS 9.4M3.

To modify an existing filter, use the command sas-display-backup-config to obtain the ID of the source instance whose filter you want to modify. Then open sas-display-backup-config.json, and locate the statements containing the name-and-value pairs that define the current filter. Copy these statements into a new JSON file, and make any needed changes. (See the JSON file specifications and examples in the previous topics.)

Then submit the sas-update-backup-config command with the -sourcetype, -id, and -inputdata options.

For example, suppose sas-display-backup-config shows the following filter to exclude log files:

```
sourceType:       configDirectories
configurableServers
  id:      tier1
  name:    tier1
  sasTierId:        tier1
  configurablePath:      C:\SAS\Config\Lev1
    included:          true
      filter
        includes
          C:\SAS\Config\Lev1\SASApp/Data, C:\SAS\Config\Lev1\SASApp/SASEnvironment
        excludes
          **/*.log
```

The following JSON file modifies the filter so that the log files are no longer excluded:

```
{
   "name": "tier1",
   "sasTierId": "tier1",
   "configurablePath": "c:\\sas\\config\\Lev1",
   "included": true,
   "filter": {
       "includes": [
         "c:\\sas\\config\\Lev1\\SASApp\\Data",
```

```
            "c:\\sas\\config\\Lev1\\SASApp\\SASEnvironment"
          ]
        }
    }
```

Here is an example command for modifying a filter for configuration directories:

```
sas-update-backup-config -profile environnment.properties -sourcetype configdirectories
-id tier1 -inputdata conf/sample/sas-update-backup-config.json
```

Delete an Existing Filter

Note: This capability is new with SAS 9.4M3.

When a filter is no longer needed, use the command sas-display-backup-config to obtain
the ID of the source instance whose filter you want to delete. Then submit the command
sas-update-backup-config with the -sourcetype, -id, and -d options. No JSON file is
needed.

Note: You cannot delete filters for the SASTIERS and CONFIGDIRECTORIES source
   types. Instead, modify the filter.

Note: You can delete only one filter at a time.

For example, suppose sas-display-backup-config shows the following filter to exclude
the SAS Content Server:

```
sourceType:      contentServer
  configurableServers
  id:     1
  name:   SAS Content Server
  sasTierId:     tier1
  included:       false
```

To delete the filter so that the content server will be included in subsequent backups, you
would use the following command:

```
sas-update-backup-config -profile environnment.properties -sourcetype contentserver
-id 1 -d
```

Add Custom Directories to the Deployment Backup (sas-add-backup-customdir)

Note: .This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and
   SAS 9.4M4, the command has been deprecated. For equivalent functionality, see sas-
   update-backup-config on page 417.

Use sas-add-backup-customdir to specify custom directories that are to be backed up (in
addition to the directories that are included by default). Each of the directories must be
located under **SAS-configuration-directory/Levn** on a host machine that is
included in the backup. The specified directories are included in all subsequent
scheduled and ad hoc executions of sas-backup.

The command has the following syntax:

sas-add-backup-customdir *web-connection-options* -inputdata *input-data-pathname*

The following list provides details:

web-connection-options

See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-inputdata *input-data-pathname*

specifies the path and filename of a JSON file that contains input data for the command. The file specifications are as follows. These specifications are effective with SAS 9.4M2.

Labels and Values	Description
"customDirectories":	Introduces a nested list of host names and associated directories that are to be included in the backup.
"hostName": "*host-name*"	Specifies the host name of a machine that contains directories that are to be included in the backup. The Deployment Backup and Recovery tool must be configured on each of these machines.
"directories":	Introduces a list of directory IDs and associated directory path values for the specified host name.
"id": "*directory-id*"	Specifies a unique identifier for this custom directory. This parameter is new with SAS 9.4M2.
"directoryPath": "*absolute-path*"	Specifies an absolute path to a directory that is to be backed up. Each of the paths must be located under **SAS-configuration-directory/ Levn**.

Here is an example input file:

```
{
"customDirectories" :
    [
        {
            "hostName": "myhost.company.com",
            "directories":
            [
                {
                    "id": "Data2",
                    "directoryPath": "c:\\sas\\config\\lev1\\Data2"
                }
            ]
        }
    ]
}
```

A sample input file is provided at **SAS-installation-directory/ SASPlatformObjectFramework/9.4/tools/admin/conf/sample/sas-add-backup-customdir.json**.

Note: Custom directories that you specify in this command add to (rather than replace) custom directories that you previously added. If you want to discontinue backing up previously added directories, use sas-remove-backup-customdir.

Note: If a custom directory that you specify does not exist when a backup is executed, the directory is ignored and no error message is generated.

Display Custom Directories for the Deployment Backup (sas-display-backup-customdir)

Note: This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. For equivalent functionality, see sas-display-backup-config on page 414.

Use sas-display-backup-customdir to display a list of additional directories that have been specified for the deployment backup using the sas-add-backup-customdir command. The command has the following syntax:

sas-display-backup-customdir *web-connection-options*

For *web-connection-options*, see "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

Here is an example of the output from this command:

```
customDirsList
        hostName:    hostmachine2
        directories
                directoryPath:  c:\sas\config\lev1\Data3
```

Remove Custom Directories from the Deployment Backup (sas-remove-backup-customdir)

Note: This command has been removed effective with SAS 9.4M5. For SAS 9.4M3 and SAS 9.4M4, the command has been deprecated. For equivalent functionality, see sas-update-backup-config on page 417.

Use sas-remove-backup-customdir to discontinue backing up a custom directory that was previously added using sas-add-backup-customdir. The specified directory will be excluded from all subsequent scheduled and ad hoc executions of sas-backup.

The command has the following syntax:

sas-remove-backup-customdir *web-connection-options* -customdirid *custom-directory-ID*

Note: This syntax is effective with SAS 9.4M2. Previously, the command required a JSON input file instead of a custom directory ID.

The following list provides details:

web-connection-options
See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-customdirid *custom-directory-ID*
specifies the ID of the custom directory that you want to delete. This is the unique identifier that you assigned to the directory as part of the input data for sas-add-backup-customdir.

Display a Deployment Recovery (sas-display-recovery)

Use sas-display-recovery to display details about a particular recovery that has been performed.

TIP The SAS Backup Manager user interface also displays details about recoveries.

The command has the following syntax:

sas-display-recovery *web-connection-options* -recoveryid *recovery-ID*

The following list provides details:

web-connection-options
See "Common Options for the Deployment Backup and Recovery Batch Commands" on page 405.

-recoveryid *recovery-ID*
specifies the identifier that was assigned to the recovery, such as `2013-03-26T13_22_41`. (To obtain the identifier, use sas-list-backups.)

Performing a Recovery Using the Deployment Backup and Recovery Tool

About Recoveries

The sas-recover-offline command restores files and databases from the backup location to all of the host machines that were included in the backup. By issuing a single command from any one of the host machines, you can restore all of the backup sources on all of the host machines in one operation.

As a best practice, you should include all backup sources in the recovery. If needed, however, you can exclude a particular source type, or you can choose to recover only specific instances of the SAS Web Infrastructure Data Server.

Effective with SAS 9.4M2, the recovery process uses checksums to validate the contents of the specified backup directory against the originally created backup. If the validation fails, a warning message is generated and the recovery is not performed. If necessary, the `forced` option can be specified in the recovery command's JSON file to perform the recovery without validating the files.

Steps for Performing a Recovery

Summary of Recovery Steps
To perform a recovery with the Deployment Backup and Recovery tool, follow these steps:

1. Identify the appropriate backup.

2. Verify that backup files are present.

3. Determine which backup sources to restore.

4. Stop all servers and services except the SAS Metadata Server, SAS Web Infrastructure Platform Data Server (all instances, including all solution data servers, as well), and SAS Deployment Agent (all instances).

5. Execute the sas-recover-offline command.

6. Perform manual steps to complete the SAS Content Server recovery (if your deployment is not current with SAS 9.4M3.).

7. Restart the servers and validate your deployment.

Note: To recover a clustered metadata server, follow the instructions in "Recovering a Clustered Metadata Server" on page 223 and then use sas-recover-offline to recover the other backup sources (excluding the metadataserver).

Each of these steps is described in the following topics.

Step 1: Identify the Appropriate Backup

First, identify the appropriate backup from which to restore your files. Follow these steps:

1. Run sas-list-backup to display the available backups.

2. From the list, identify a backup from which to recover. Use the following criteria:

   - Select the most recent backup that was taken before the issue that prompted the recovery.

   - If you recently upgraded to a new version of SAS software, make sure that the backup was taken after the upgrade. Restoring content from a previous SAS version could cause problems.

   - If you recently used SAS Deployment Manager to update host name references, make sure that the backup was taken after the update.

3. Run sas-display-backup to obtain details about the selected backup, and check the output as follows:

   - Make sure that the overall status of the backup is not **Offline** (expired).

   - Make sure that each backup source (METADATASERVER, CONTENTSERVER, DATABASE, CONFIGDIRECTORIES, and CUSTOMDIRECTORIES, if applicable) has a status of **complete**.

Step 2: Verify That Backup Files Are Present

Verify that the backup files are present in the backup location.

If a central vault location has been defined, do the following:

1. Make sure that the central vault location contains a backup directory for the selected backup. The directory name must be exactly the same as the backup ID displayed by sas-display-backup.

2. Make sure that the backup directory contains a subdirectory for each of the backup sources and that each of the subdirectories contains backup files.

If a central vault has not been defined, do the following:

1. On each host machine that is included in the backup, make sure that the backup location (for example, *SAS-configuration-directory*/**Lev1/Backup/ Vault**) contains a backup directory for the selected backup. The directory name must be exactly the same as the backup ID displayed by sas-display-backup.

2. On each host machine, make sure that the backup directory contains a subdirectory for the appropriate backup sources and that each of the subdirectories contains backup files.

Step 3: Determine Which Backup Sources to Restore

By default, the recovery command restores all of the source types (METADATASERVER, CONTENTSERVER, DATABASE, CONFIGDIRECTORIES, and CUSTOMDIRECTORIES, if applicable). Generally, you should restore all of these source types to keep your content in sync.

However, there might be situations in which you need to exclude a particular source type, or in which you need to recover only specific instances of the SAS Web Infrastructure Data Server. To do so, update the JSON file for the sas-recover-offline command before you run the command. For details, see "Command Reference for sas-recover-offline " on page 433.

Step 4: Stop All Servers and Services except the Metadata Server, Web Infrastructure Platform Data Server, and Deployment Agent

Keep at least the following servers running:

1. SAS Metadata Server

2. SAS Web Infrastructure Platform Data Server (all instances)

3. SAS Deployment Agent (all instances)

Note: You might need to keep additional servers running depending on what products you have installed.

Stop all of the other SAS servers and services in your deployment.

Note: These instructions assume that you will be using sas-recover-offline, which is the supported method to perform recoveries.

Step 5: Execute sas-recover-offline

Execute the command sas-recover-offline using the syntax described in "Command Reference for sas-recover-offline " on page 433.

You can launch the command from any host machine that is included in the backup. The command restores files to all of the applicable host machines.

Monitor the messages that are written to the console to ensure that the recovery completes successfully. If errors or warning are displayed, see "Resolving Recovery Issues" on page 436.

Step 6: Perform Manual Steps to Complete SAS Content Server Recovery (If Deployment Is Not Current with Third Maintenance Release)

Note: If your deployment is current with SAS 9.4M3, then omit this step. The SAS Content Server recovery will be completed automatically.

If you are recovering the SAS Content Server repository, and if your deployment is not current with SAS 9.4M3, then you must perform the following additional manual steps after executing the sas-recover-offline command:

1. Stop the SAS Web Application Server.

2. Navigate to **SAS-configuration-directory/Lev1/AppData/ SASContentServer**. The **SASContentServer** directory should contain two

directories: `Copy_of_Repository` and `Repository`. The date and time on the `Copy_of_Repository` directory should be soon after the time at which the recovery operation completed.

3. Back up the current `Repository` directory by renaming it to a name such as `Repository_Old`. Then rename `Copy_of_Repository` to `Repository`.

4. To complete the synchronization of the database, follow these steps:

   a. Navigate to *SAS-installation-directory/* ***SASWebInfrastructurePlatformDataServer/9.4/bin***.

   b. On UNIX machines only, set the LD_LIBRARY_PATH environment variable as follows:

      i. Enter the following statement:

LD_LIBRARY_PATH=*SAS-installation-directory*/SASWebInfrastructurePlatformDataServer/9.4/lib:$LD_LIBRARY_PATH

         For example:

LD_LIBRARY_PATH=usr/local/SASHome/SASWebInfrastructurePlatformDataServer/9.4/lib:$LD_LIBRARY_PATH

      ii. Run the following command:

          export LD_LIBRARY_PATH

   c. On all platforms, run the following command:

      `psql -h host-name -p port -d SharedServices -U database-owner`

      Note: The command options are case-sensitive.

      Provide the following values in the command:

-h *host-name*	The fully qualified name of the host machine where the SAS Web Infrastructure Platform Data Server is installed.
-p *port*	The port on which the server is accessed. The default port number is 9432. (You might have specified a different port number when the SAS Deployment Wizard was run.)
-U *database-owner*	The user that owns the database. The default name for this user is **dbmsowner**. The user is specified in the following file: *SAS-configuration-directory*/Lev1/Web/ WebAppServer/SASServer1_1/conf/server.xml Look for the line that contains the following text: **driverClassName="org.postgresql.Driver"**

   d. When prompted, enter the password for the database owner.

   e. When the **SharedServices** prompt appears, enter the following statement:

      `insert into SAS_SCS2_GLOBAL_REVISION VALUES(0,-1);`

   f. At the next prompt, enter **\q** to quit the psql program.

5. Restart the SAS Web Application Server.

6. After the SAS Content Server is online, you can delete the old repository folder that you renamed (for example, `Repository_Old`).

Step 7: Restart Servers and Validate Your Deployment

When the recovery has been successfully completed, follow these steps:

1. Start all SAS servers and servers. See "Overview of Server Operation" on page 62.

2. Before making the deployment available to users, validate your SAS servers and applications. For validation details, see the instructions.html document in **SAS-installation-directory/Lev1/Documents** on each host machine that was included in the recovery.

Command Reference for sas-recover-offline

Use sas-recover-offline to restore files from a deployment backup. This command, which is new with SAS 9.4M1, is the supported method for performing recoveries.

This command works when some of the resources in the deployment are unavailable or have been taken offline to prevent user activity. Only the following resources need to be running:

• metadata server

• SAS Deployment Agent (all instances)

• SAS Web Infrastructure Platform Data Server (all instances)

Stop all of the other SAS servers and services in your deployment.

Note: If you have specified a central vault location, the backup files are copied from the central vault location to the local backup locations before the recovery is performed.

Note: If you are recovering the instance of the SAS Web Infrastructure Data Server that includes the EVManager database, the SAS Environment Manager server must be stopped.

Note: To recover a clustered metadata server, follow the instructions in "Recovering a Clustered Metadata Server" on page 223.

The command syntax is as follows:

sas-recover-offline -host *metadata-server-host-name* -port *metadata-server-port*

-user *user-id* -password *password* -profile *filename*

<-inputdata *input-data-pathname*> -backupid *backup-ID*

`T I P`   Be sure to provide connection information for the metadata server—not for the web server or the web application server.

The following list provides details:

-host *metadata-server-host-name*
identifies the host machine for the metadata server. This option is required if the -profile option is not set.

-port *metadata-server-port*
specifies the port on which the metadata server runs. This option is required if the -profile option is not set.

-user *user-id*
specifies the user ID of an unrestricted user (for example, sasadm@saspw). This option is required if the -profile option is not set.

-password *password*
> specifies the password of the specified user. This option is required if the -profile option is not set. The password should be encrypted using SAS proprietary 32-bit encryption. To obtain the encrypted password, use PROC PWENCODE.

-profile *filename*
> specifies the name of the connection profile that is to be used to connect to the metadata server. The connection profile must exist on the computer where the command is executed and must specify the credentials of the SAS Administrator user. You can specify a connection profile that has been created for use with client applications such as SAS Management Console, SAS Data Integration Studio, and SAS OLAP Cube Studio. When you open one of these applications, the available connection profiles are displayed in the drop-down box in the Connect Profile dialog box.
>
> This option can be provided in place of -host, -port, -user, and -password.

-inputdata *input-data-pathname*
> specifies the path and filename of a JSON file that contains input data for the command. The input data is not required for this command. The file specifications are as follows:

Labels and Values	Description
"name": "*restore-name*"	Specifies a name to identify this restore in backup and restore history. The name must be unique.
"comment": "*comment*"	Specifies a comment that is to be stored in the backup and restore history.
"pauseComment": "*pause-comment*"	Specifies a comment that is to be displayed to users when the servers are paused during the recovery.
"instanceName":	Specifies instances of the SAS Web Infrastructure Platform Server that are to be included in the recovery. Enter the server name exactly as it appears in the **Server Manager** tree in SAS Management Console. If you specify multiple instances, separate them with double commas.
	If you do not specify this option, then all instances are included in the recovery.
	Note: When using this option in a deployment that is not current with SAS 9.4M2 or SAS 9.4M3, you must also use the **"exclude":** parameter to exclude all sources except **database**.
"exclude":"metadataserver" | **"contentserver"** | **"database"** | **"configdirectories"** | **"customdirectories"**	Specifies backup sources that are to be excluded from the recovery.
	Note: Use caution when excluding sources. Doing so can cause your data to be out of sync.
	Note: The labels are effective with SAS 9.4M3. In earlier releases, specify **"configdir"** instead of **"configdirectories"**, and specify **"customdir"** instead of **"customdirectories"**.

Labels and Values	Description
`"forced": "true"|"false"`	Specifies **`"true"`** or **`"false"`** to indicate whether the recovery is to be executed even if the backup files that are being used cannot be successfully validated. This option is new with SAS 9.4M2. The default setting is **`"false".`**
`"includeMetadataConfig" : "true"|"false"`	*Note:* This option is available with SAS 9.4M3. Specifies **`"true"`** or **`"false"`** to indicate whether the metadata server's configuration files are to be recovered. If you specify **`"false"`**, only the metadata repositories and the repository manager are included. The default setting is **`"false".`**
`"reorganizeMetadataRepositories" : "true"|"false"`	*Note:* This option is available with SAS 9.4M3. Specifies **`"true"`** or **`"false"`** to indicate whether the metadata repositories are to be reorganized during the recovery to reclaim unused disk space. The default setting is **`"false".`**

Here is a sample input file:

Note: This example is valid for SAS 9.4M3. In earlier releases, specify **`"configdir"`** instead of **`"configdirectories"`**, and specify **`"customdir"`** instead of **`"customdirectories"`**.

```
{
    "name": "Restore",
    "comment": "Restore of metadata server, content server, and selected
                instances of the Web Infrastructure Platform Database",
    "pauseComment": "pausecomment",
    "instanceName": "Web Infrastructure Platform DataServer 94,,
                    Fin Mgmt Data Svr 5.4"
    "exclude": ["contentserver","configdirectories","customdirectories"],
        "includeMetadataConfig": "true"    }
```

-backupid *backup-ID*

specifies the identifier of the backup from which to recover. The identifier is the same as the name of backup directory (for example, `2013-02-01T03_13_01`).

CAUTION:

If you are recovering the SAS Content Server repository, and if your deployment is not current with SAS 9.4M3, you must perform additional manual steps after executing the offline recovery command. See "Step 6: Perform Manual Steps to Complete SAS Content Server Recovery (If Deployment Is Not Current with Third Maintenance Release)" on page 431.

As a best practice, you should restart all of your servers when the recovery is complete.

Command Reference for sas-recover

Use sas-recover to restore files from the deployment backup locations on each host machine.

Note: Effective with SAS 9.4M1, sas-recover-offline is the supported method for performing a recovery. See "Command Reference for sas-recover-offline " on page 433. The sas-recover command continues to be available for SAS 9.4TS1M0 only.

Resolving Recovery Issues

Checksum Validation Error

Effective with SAS 9.4M2, the recovery process uses checksums to validate the contents of the specified backup directory against the originally created backup. If the validation fails, a warning message is generated and the recovery is not performed.

Follow these steps to determine the cause of the validation error:

1. Check the backup directory to determine whether it contains a file called backupsourceDetails.checksum.

   If the file is not present, this means that a checksum calculation error might have occurred during the backup process. In this situation, you can attempt to perform a forced recovery. To do so, specify the **forced** option in the JSON file for the recovery command.

2. If the backup directory contains a file called backupsourceDetails.checksum, then a validation error indicates that the backup files have changed since the backup was created. In this situation, a forced recovery is not recommended. You might want to consider recovering from a different backup.

Failure to Restore a Specific Backup Source

If validation is successful and the recovery proceeds, an error could still occur during the recovery operation. Follow these steps:

1. Review the console messages to determine which backup sources and which instances of the SAS Web Infrastructure Platform Data Server were successfully restored and which ones could not be restored.

2. Review the console messages and the backupserver.log located at **SAS-configuration-directory/Lev1/Backup** to determine the cause of the error. Take steps to resolve the error situation.

   For example, a database recovery error could be caused by a failure to connect to one of the databases on the SAS Web Infrastructure Platform Database Server. A SAS Content Server recovery error could be caused by failure to connect to the SharedServices database on this server. To resolve these types of errors, make sure that the SAS Web Infrastructure Platform Database Server instance that contains the database has been started.

3. Update the JSON file to exclude backup sources and data server instances that have already been successfully restored.

4. Attempt to run the recovery command again with the updated JSON file.

Part 9

Appendixes

Chapter 29
Reference: Configuration Directories

Introduction to the Configuration Directory Reference

This appendix contains details about the contents of the main subdirectories of the SAS Intelligence Platform configuration directory, which exists on each server-tier and middle-tier host machine. The configuration directory is referred to in the documentation as *SAS-configuration-directory* and is located in the following path:

* on Windows systems:

  drive:\SAS\configuration-directory-name

* on UNIX systems:

  installer's-home-directory/SAS/configuration-directory-name

* on z/OS systems, the directory that is specified in the **CONFIG_DIR** environment variable

In these paths, *configuration-directory-name* is the directory name that was specified when the deployment plan was created. If no name was specified, then the default name **Config** is used. During installation, the SAS Deployment Wizard enables you to override the configuration directory path with a different path.

For an overview of the configuration directory structure, see "Overview of the Configuration Directory Structure" on page 14.

Contents of the Levn Subdirectory

The `Levn` subdirectory contains configuration information and other files for a particular installation instance. `Lev1` is generally used for production environments, and additional levels, such as `Lev2` and `Lev3`, can be used for environments that you install for purposes such as development and testing. During installation, the SAS Deployment Wizard enables you to specify the level number.

Depending on what software is installed on the machine, the `Levn` directory can contain the subdirectories and files that are listed in the following table:

Table 29.1 *Contents of the Levn Subdirectory*

Subdirectory or File	Description
`AppData`	Contains indexes and the repository configuration file for the SAS Content Server, and data that is installed for the use of specific applications (for example, SAS BI Dashboard).
`Backup/Vault`	Is the default location for backups that are created by the Deployment Backup and Recovery Tool.
`ConnectSpawner`	Contains the management script, configuration files, and logs for the SAS/CONNECT spawner.
`Data`	Can be used to store user data.
`DeploymentTesterServer`	Contains files that are used by the Deployment Tester plug-in to SAS Management Console.
`Documents`	Contains Instructions.html, which contains post-installation configuration instructions; DeploymentSummary.html; ConfigurationErrors.html; and other application-specific documents.
`Logs`	Can be used as a common directory for server and spawner logs, if you selected this option during a custom installation. By default, each server has its own separate log directory. See "Default Locations for Server Logs" on page 445.
`Logs/Configure`	Contains logs that are created by the SAS Deployment Wizard.
`ObjectSpawner`	Contains a management script, configuration files, and logs for the object spawner.
`SASApp`	Contains management scripts, configuration files, and logs for SAS Application Server components. See "Contents of the Application Server Subdirectory (SASApp)" on page 443.
`SASMeta`	Contains management scripts, configuration files, metadata repositories, logs, and other files and directories for the SAS Metadata Server. See "Contents of the Metadata Server Subdirectory (SASMeta/MetadataServer)" on page 441.

Subdirectory or File	Description
ShareServer	Contains the management script, configuration information, and log files for the SAS/SHARE server.
Utilities	Contains XML files that are used as input to the SAS Deployment Wizard and utilities that are associated with this configuration instance.
Web	See "Contents of the Web Subdirectory" on page 444.
WebInfrastructurePlatformDataServer	Contains the management script, configuration information, and log files for the SAS Web Infrastructure Platform Data Server.
generate_boot_scripts.sh	Is a script that is used to regenerate the sas.servers script on UNIX and z/OS. See "Regenerating a sas.servers Script" on page 82.
metadataConfig.xml	Provides information for application server components to use when they connect to a clustered metadata server.
sas.servers	Is a management script that is used on UNIX and z/OS to start, stop, or restart all servers on the machine in the correct order, or to display the status of all servers on the machine.
sasv9_meta.cfg	Specifies metadata server connection information for the SAS OLAP Server and SAS/SHARE server.

Contents of the Metadata Server Subdirectory (SASMeta/MetadataServer)

On the machines where the metadata server is installed, the **Levn** directory includes a subdirectory for the metadata server's files. **SASMeta** is the name that the SAS Deployment Wizard assigns to the metadata server context and, in turn, to this directory.

The **SASMeta** directory contains a **MetadataServer** subdirectory. The following table lists the contents of this subdirectory. For more information about the metadata server configuration files, see "Configuration Files for the Metadata Server" on page 448.

Table 29.2   Contents of the MetadataServer Subdirectory of SASMeta

Subdirectory or File	Description
Backups	Is the default location for backups of the metadata server.
Journal	Contains the metadata server journal files. For details, see "Configuring Metadata Server Journaling" on page 202.
Logs	Is the default location for metadata server logs.

Subdirectory or File	Description
MetadataRepositories	Contains subdirectories for each metadata repository. These subdirectories contain the SAS data sets that contain your metadata.
rposmgr	Contains data sets for the repository manager.
sasuser	Is a SAS library that can be used to store SAS catalogs that enable you to customize certain features of SAS to meet specific needs. SAS assigns the SASUSER library at invocation.
adminUsers.txt, trustedUsers.txt, and trustedPeers.xml	Contain security-related configuration information. See "About the Security-Related Configuration Files" on page 18.
backupServer.sas	Is invoked by MetadataServer.bat when the backup option is used. This program uses the metadata server backup facility to back up the metadata server repositories, repository manager, and server configuration information. For details, see Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169.
logconfig.xml	Specifies the logging configuration for the metadata server. For details, see Chapter 10, "Administering Logging for SAS Servers," on page 131.
MetadataServer.bat or MetadataServer.sh	Is a script that is used to stop, start, pause, resume, or report the status of the metadata server.
MetadataServer.srv	On UNIX and z/OS, contains information that is used by generate_boot_scripts.sh to regenerate the sas.servers script.
MetadataServerBackupManifest.xml	Contains detailed information about the most recent metadata server backup. *Note:* Do not delete or move this file from the metadata server directory. Doing so prevents the server from starting successfully.
MetadataServerSSCU.ini	Configures the metadata server service in the Windows Services Manager.
MetadataServer_shortcuts.ini	Configures shortcuts for the Windows **Start** menu.
metaparms.sas	Contains user credentials that can be used to execute SAS sessions that include PROC METAOPERATE or PROC METADATA.
omaconfig.xml	Is the SAS Metadata Server configuration file.
optimizeIMDB.sas	Is invoked by MetadataServer.bat or MetadataServer.sh when the optimizeIMDB option is used. See "Using optimizeIMDB to Optimize and Report on Metadata Server Memory Usage" on page 208.

Subdirectory or File	Description
pauseServer.sas, resumeServer.sas, statusServer.sas, and stopServer.sas	Are programs that are invoked by MetadataServer.bat or MetadataServer.sh when the pause, resume, status, and stop options are used.
sasv9.cfg	Specifies SAS options that are put into effect when the metadata server is started.
sasv9_usermods.cfg	Contains user modifications to the SAS options that are specified in sasv9.cfg.

Contents of the Application Server Subdirectory (SASApp)

On machines where components of a SAS Application Server are installed, the **Levn** directory contains an application server subdirectory. A SAS Application Server is a logical entity that contains specific servers (for example, a SAS Workspace Server and a SAS Stored Process Server) that execute SAS code. Application server subdirectories are named as follows:

- **SASApp** is the default name that the SAS Deployment Wizard assigns to directory for the first application server. If you assigned a different name when you ran the SAS Deployment Wizard, then the subdirectory is named accordingly.

- If you configure more than one application server, then the SAS Deployment Wizard creates additional directories that correspond to each of the application server names.

The **SASApp** subdirectory contains the following items:

- several files that control the application server configuration. For details, see "Reference: Configuration Files for SAS Servers" on page 447.

- some or all of the following subdirectories, depending on which SAS servers have been configured.

  - **BatchServer**

  - **ConnectServer**

  - **GridServer**

  - **OLAPServer**

  - **PooledWorkspaceServer**

  - **StoredProcessServer**

  - **WorkspaceServer**

These subdirectories contain start-up scripts, configuration files, logconfig.xml files to configure logging, and **Logs** directories to store logs. For details about the configuration files, see "Reference: Configuration Files for SAS Servers" on page 447. For details about start-up scripts, see Chapter 6, "Operating Your Servers," on page 61.

- a **Data** subdirectory, which you can use to store data to be used by components of this application server.

- a **SASEnvironment** subdirectory for storing the following elements of the run-time environment for SAS code that runs on the application server:

  - a **SASCode** subdirectory, which you can use as the deployment directory for SAS Data Integration Studio jobs that are deployed for scheduling. This directory contains a **Jobs** subdirectory to store jobs and a **Steps** subdirectory to store the SAS code for each job step. These directories enable source code to be shared among developers on a project.

  - a **SASFormats** subdirectory where you can store SAS format and informat catalogs for use by applications, such as SAS Data Integration Studio and SAS Information Map Studio, that execute code on the application server.

  - a **SASMacro** subdirectory, where you can store user-created SAS autocall macros that are invoked by SAS code running on the application server.

Contents of the Web Subdirectory

On middle-tier machines (that is, machines with installed web components), the **Lev1** directory contains a subdirectory called **Web** that contains these items:

- an **Applications** subdirectory, which contains a subdirectory for each of the web applications and other middle-tier applications that are installed on the machine. These directories contain configuration files, policy files, and other support files for the applications.

- a **Common** subdirectory, which contains configuration files for services that are shared by web applications.

- a **Logs** subdirectory, which contains logs for web applications.

- a **Staging** subdirectory, which contains the enterprise archive (EAR) files for your web applications.

- a **Utilities** subdirectory, which contains utilities related to installed web applications.

- the **activemq**, **geode**, **SASEnvironmentManager**, **WebAppServer**, and **WebServer** directories, which contain configuration information, logs, and other files for the SAS middle-tier servers and related services.

  Note: For releases prior to SAS 9.4M8, the gemfire directory (not the geode directory) contains the content.

- a **Scripts** subdirectory, which contains management scripts for the SAS middle-tier servers.

For information about administering middle-tier applications and components, see the *SAS Intelligence Platform: Middle-Tier Administration Guide* and the *SAS Intelligence Platform: Web Application Administration Guide*.

Chapter 30
Default Locations for Server Logs

Default Locations for Server Logs

The following table shows the default location of server logs within the SAS configuration directory:

Note:

- **SASApp** is the default name that the SAS Deployment Wizard assigns to directory for the first application server. If you assigned a different name when you ran the SAS Deployment Wizard, then the subdirectory is named accordingly.

- In a custom installation, you can choose to place all of your server-tier log files in a single directory. If this option was selected, then your server-tier log files are written to the path that was specified during installation (for example, *SAS-configuration-directory*/Lev1/Logs).

Table 30.1 Location of Server Logs

Server	Log Location within the *SAS-configuration-directory*
SAS Metadata Server	Windows: **Lev1\SASMeta\MetadataServer\Logs**
	UNIX or z/OS: **Lev1/SASMeta/MetadataServer/Logs**
SAS object spawner	Windows: **Lev1\ObjectSpawner\Logs**
	UNIX or z/OS: **Lev1/ObjectSpawner/Logs**
SAS OLAP Server	Windows: **Lev1\SASApp\OLAPServer\Logs**
	UNIX or z/OS: **Lev1/SASApp/OLAPServer/Logs**
SAS Pooled Workspace Server	Windows: **Lev1\SASApp\PooledWorkspaceServer\Logs**
	UNIX or z/OS: **Lev1/SASApp/PooledWorkspaceServer/Logs**

Server	Log Location within the *SAS-configuration-directory*
SAS Stored Process Server	Windows: **Lev1\SASApp\StoredProcessServer\Logs** UNIX or z/OS: **Lev1/SASApp/StoredProcessServer/Logs**
SAS Workspace Server	For this server, the initial logging configuration sends log events to the operating system facility rather than to a log file.
SAS/SHARE server	Windows: **Lev1\ShareServer\Logs** UNIX or z/OS: **Lev1/ShareServer/Logs**
SAS/CONNECT spawner	Windows: **Lev1\ConnectSpawner\Logs** UNIX or z/OS: **Lev1/ConnectSpawner/Logs**
SAS/CONNECT server	For this server, the initial logging configuration sends log events to the operating system facility rather than to a log file.
SAS DATA step batch server	Windows: **Lev1\SASApp\BatchServer\Logs** UNIX or z/OS: **Lev1/SASApp/BatchServer/Logs**
SAS Web Infrastructure Platform Data Server	Windows: **Lev1\WebInfrastructurePlatformDataServer\Logs** UNIX or z/OS: **Lev1/WebInfrastructurePlatformDataServer/Logs**

For more information about logging for servers on the SAS server tier, see Chapter 10, "Administering Logging for SAS Servers," on page 131. For information about logs for middle-tier components, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Chapter 31

Configuration Files

Reference: Configuration Files for SAS Servers

Introduction to the Configuration File Reference

This section provides tables that show the name, location, and purpose of the various server configuration files that are used in the SAS Intelligence Platform. Each table in this appendix contains the following columns:

Order of Precedence
> The numbers in this column indicate the order in which the options in the files are applied. Option settings in files with higher orders of precedence override the settings in files with lower orders of precedence.
>
> "Not applicable" means that the option settings in the file do not affect, and are not affected by, option settings that are specified in other files.

Path and Filename
> All paths that begin with **Lev1** are located in the SAS configuration directory. See "About the Configuration Directory" on page 14. Within the pathnames, the following variables are used:

- *server-context*, which indicates the name of an application server grouping (for example, **SASApp**)

- *server-name*, which indicates the name of the logical server or server that the configuration file applies to

Purpose
This column describes the purpose of the configuration file, indicates that certain configuration files are not to be modified by users, and provides references to detailed documentation.

Configuration Files for the Metadata Server

The following table shows the configuration files that affect the metadata server. For an explanation of each columns' contents, see "Introduction to the Configuration File Reference" on page 447.

Table 31.1 *Configuration Files for the SAS Metadata Server*

Order of Precedence	Path and Filename	Purpose
1	Windows: **\Lev1\SASMeta\MetadataServer \sasv9.cfg** UNIX and z/OS: **/Lev1/SASMeta/ MetadataServer/sasv9.cfg**	Specifies server start-up options for the metadata server and contains calls to the other files that are listed in the table. Do not modify this file. If you need to make changes, modify the sasv9_usermods.cfg file that is in the same directory.
2	Windows: *SAS-install-directory* **\SASFoundation\9.4\sasv9.cfg** UNIX: *SAS-install-directory***/SAS-install-directory/ SASFoundation/9.4/sasv9.cfg** z/OS: Not applicable	Specifies system options that apply to all SAS sessions on the host machine. On UNIX systems, do not modify this file. If you need to make changes, modify the sasv9_local.cfg file that is in the same directory.
3	UNIX only: *SAS-install-directory*/ **SASFoundation/9.4/sasv9_local.cfg**	On UNIX systems, contains site-specific modifications to the sasv9.cfg file that is in the same directory.
4	Windows: *SAS-install-directory* **\SASFoundation\9.4***locale***\sasv9.cfg** UNIX: *SAS-install-directory***/ SASFoundation/9.4/***locale***/sasv9.cfg** z/OS: **<HLQ>.V93D8***ddd***.CNTLDSN(BAT***locale***W0)**	Specifies system options that apply to all SAS sessions on the host machine in the specified locale.
5	Windows: **\Lev1\SASMeta\MetadataServer \sasv9_usermods.cfg** UNIX and z/OS: **/Lev1/SASMeta/ MetadataServer/sasv9_usermods.cfg**	Contains site-specific modifications to the metadata server start-up options. Use this file to change the options that are specified in the sasv9.cfg file in the same directory. See Chapter 32, "Options for the Metadata Server Invocation Command," on page 469.
Not applicable	Windows: **\Lev1\SASMeta\MetadataServer \omaconfig.xml** UNIX and z/OS: **/Lev1/SASMeta/ MetadataServer/omaconfig.xml**	Specifies options for the metadata server and metadata repositories. See "Reference Information for omaconfig.xml" on page 456.

Order of Precedence	Path and Filename	Purpose
Not applicable	Windows: `\Lev1\SASMeta\MetadataServer\logconfig.xml` UNIX and z/OS: `/Lev1/SASMeta/MetadataServer/logconfig.xml`	Specifies the logging configuration for the server. See Chapter 10, "Administering Logging for SAS Servers," on page 131.
Not applicable	Windows: `\Lev1\SASMeta\MetadataServer\adminUsers.txt` UNIX and z/OS: `/Lev1/SASMeta/MetadataServer/adminUsers.txt`	Defines your system's initial unrestricted user. See "Metadata Server: Unrestricted Role" on page 19.
Not applicable	Windows: `\Lev1\SASMeta\MetadataServer\trustedUsers.txt` UNIX and z/OS: `/Lev1/SASMeta/MetadataServer/trustedUsers.txt`	Conveys privileged status to a service account to enable that account to act on behalf of other users. See "About the Initial User Accounts" on page 22.
Not applicable	Windows: `\Lev1\SASMeta\MetadataServer\trustedPeers.xml` UNIX and z/OS: `/Lev1/SASMeta/MetadataServer/trustedPeers.xml`	Can be used to limit the scope of trust for connections from SAS sessions to the metadata server. Initially, the scope is not limited. See the *SAS Intelligence Platform: Security Administration Guide*.
Not applicable	Windows: Not applicable UNIX and z/OS: `/Lev1/SASMeta/MetadataServer/MetadataServer.srv`	On UNIX and z/OS, contains information that is used by generate_boot_scripts.sh to regenerate the sas.servers script.
Not applicable	Windows: `\Lev1\SASMeta\appservercontext_env.sh` UNIX and z/OS: `/Lev1/SASMeta/appservercontext_env.sh`	Used to define site-specific system environment variables or make operating system-level changes, such as umask of a process. Do not modify this file. If you need to make changes, modify the appservercontext_env_usermods.sh file that is in the same directory.

In z/OS environments, the following additional configuration files affect the metadata server:

Table 31.2   Additional Configuration Files for the SAS Metadata Server on z/OS Systems

Order of Precedence	Path and Filename	Purpose
1	z/OS: `<HLQ>.TKMVSENV(TKMVSENV)`	Contains name/value pairs (environment variables) that apply to all SAS sessions on the machine.
2	z/OS: `/Lev1/SASMeta/MetadataServer/tkmvsenv.cfg`	Contains name/value pairs (environment variables) that apply to the metadata server. Do not modify this file. If you need to make changes, modify the tkmvsenv_usermods.cfg file that is in the same directory.

Order of Precedence	Path and Filename	Purpose
3	z/OS: **/Lev1/SASMeta/MetadataServer/ tkmvsenv_usermods.cfg**	Contains site-specific modifications to the name/ value pairs (environment variables) for the metadata server. Use this file to change the options that are specified in the tkmvsenv.cfg file in the same directory. See the *Configuration Guide: SAS Foundation for z/OS*.

Configuration Files for Components of SAS Application Servers

The following table shows the configuration files that affect servers that are components of a SAS Application Server. These servers include OLAP servers, workspace servers, pooled workspace servers, stored process servers, and SAS/CONNECT servers. For an explanation of each columns' contents, see "Introduction to the Configuration File Reference" on page 447.

Table 31.3 *Configuration Files for Components of SAS Application Servers (OLAP Servers, Workspace Servers, Stored Process Servers, and SAS/CONNECT Servers)*

Order of Precedence	Path and Filename	Purpose
1	Windows: **\Lev1*server-context**server-name*\sasv9.cfg** UNIX and z/OS: **/Lev1/*server-context*/ *server-name*/sasv9.cfg**	Specifies start-up options for the server and contains calls to the other files that are listed in this table. Do not modify this file. If you need to make changes, modify the sasv9_usermods.cfg file that is in the same directory.
2	Windows: **\Lev1*server-context*\sasv9.cfg** UNIX and z/OS: **/Lev1/*server-context*/ sasv9.cfg**	Specifies start-up options that apply to all components of the application server. Do not modify this file. If you need to make changes, modify the sasv9_usermods.cfg file that is in the same directory. See "Overview of System Options for SAS Application Server Components" in *SAS Intelligence Platform: Application Server Administration Guide*.
3	Windows: ***SAS-install-directory* \SASFoundation\9.4\sasv9.cfg** UNIX: ***SAS-install-directory*/ SASFoundation/9.4/sasv9.cfg** z/OS: Not applicable	Specifies system options that apply to all SAS sessions on the host machine. On UNIX systems, do not modify this file. If you need to make changes, modify the sasv9_local.cfg file that is in the same directory.
4	UNIX only: ***SAS-install-directory*/ SASFoundation/9.4/sasv9_local.cfg**	On UNIX systems, contains site-specific modifications to the sasv9.cfg file that is in the same directory.

Order of Precedence	Path and Filename	Purpose
5	Windows: *SAS-install-directory* **\SASFoundation\9.4***locale***\sasv9.cfg** UNIX: *SAS-install-directory/* **SASFoundation/9.4/***locale***/sasv9.cfg** z/OS: **<HLQ>.V93D8***ddd***.CNTLDSN(BAT***locale***W0)**	Specifies system options that apply to all SAS sessions on the host machine in the specified locale.
6	Windows: **\Lev1***server-context* **\sasv9_usermods.cfg** UNIX and z/OS: **/Lev1/***server-context***/ sasv9_usermods.cfg**	Contains site-specific modifications to the server start-up options that apply to all application server components. Use this file to change the options that are specified in the sasv9.cfg file in the same directory. See "Overview of System Options for SAS Application Server Components" in *SAS Intelligence Platform: Application Server Administration Guide*.
7	Windows: **\Lev1***server-context* **\appserver_autoexec.sas** UNIX and z/OS: **/Lev1/***server-context***/ appserver_autoexec.sas**	Contains SAS statements that are executed immediately after SAS initializes on all components of the application server. Do not modify this file. If you need to make changes, modify the appserver_autoexec_usermods.sas file that is in the same directory.
8	Windows: **\Lev1***server-context* **\appserver_autoexec_usermods.sas** UNIX and z/OS: **/Lev1/***server-context***/ appserver_autoexec_usermods.sas**	Contains site-specific modifications to the appserver_autoexec.sas file that is located in the same directory.
9	Windows: **\Lev1***server-context******server-name***\sasv9_usermods.cfg** UNIX and z/OS: **/Lev1/***server-context***/ server-name/sasv9_usermods.cfg**	Contains site-specific modifications to the server's start-up options. Use this file to change the options that are specified in the sasv9.cfg file in the same directory. See "Overview of System Options for SAS Application Server Components" in *SAS Intelligence Platform: Application Server Administration Guide*.
10	Windows: **\Lev1***server-context******server-name***\autoexec.sas** UNIX and z/OS: **/Lev1/***server-context***/ server-name/autoexec.sas**	Contains SAS statements that are executed immediately after SAS initializes on the server. Do not modify this file. If you need to make changes, modify the autoexec_usermods.sas file that is in the same directory.
11	Windows: **\Lev1***server-context******server-name***\autoexec_usermods.sas** UNIX and z/OS: **/Lev1/***server-context***/ server-name/autoexec_usermods.sas**	Contains site-specific modifications to the autoexec.sas file that is located in the same directory.
Not applicable	Windows: UNIX and z/OS: **/Lev1/***server-context***/ server-name/logconfig.xml**	Specifies the logging configuration for the server. See Chapter 10, "Administering Logging for SAS Servers," on page 131.

Order of Precedence	Path and Filename	Purpose
Not applicable	Windows: **\Lev1\sasv9_meta.cfg** UNIX and z/OS: **/Lev1/sasv9_meta.cfg**	Specifies metadata server connection information for the SAS OLAP Server and SAS/SHARE server.
Not applicable	Windows: **\Lev1\metadataConfig.xml** UNIX and z/OS: **/Lev1/metadataConfig.xml**	Provides information for application server components to use when they connect to a clustered metadata server.
Not applicable	Windows: Not applicable UNIX and z/OS: **/Lev1/*server-context*/*server-name*/*server-name*.srv**	On UNIX and z/OS, contains information that is used by generate_boot_scripts.sh to regenerate the sas.servers script.
Not applicable	Windows: **\Lev1\SASApp\appservercontext_env.sh** UNIX and z/OS: **/Lev1/SASApp/appservercontext_env.sh**	Used to define site-specific system environment variables or make operating system-level changes, such as umask of a process. Do not modify this file. If you need to make changes, modify the appservercontext_env_usermods.sh file that is in the same directory.
Not applicable	Windows: **\Lev1\level_env.sh** UNIX and z/OS: **/Lev1/level_env_usermods.sh**	Used to set environment variables relative to the configuration level structure. Do not modify this file. If you need to make changes, modify the level_env_usermods.sh file that is in the same directory.
Not applicable	Windows: **\Lev1*server-context**server-name**server-name*.sh** UNIX and z/OS: **/Lev1/*server-context*/*server-name*/*server-name*_usermods.sh**	Invokes SAS with the default configuration for the respective site-specific application server. It changes directories so that SAS is invoked from the root directory of the site-specific application server. If you need to make changes, modify the server-name_usermods.sh file that is in the same directory.

In z/OS environments, the following additional configuration files affect components of a SAS Application Server:

Table 31.4 *Additional Configuration Files for Components of SAS Application Servers (OLAP Servers, Workspace Servers, Stored Process Servers, and SAS/CONNECT Servers) on z/OS Systems*

Order of Precedence	Path and Filename	Purpose
1	z/OS: **<HLQ>.TKMVSENV(TKMVSENV)**	Contains name/value pairs (environment variables) that apply to all SAS sessions on the machine.
2	z/OS: **/Lev1/*server-context*/*server-name*/tkmvsenv.cfg**	Contains name/value pairs (environment variables) that apply to this server. Do not modify this file. If you need to make changes, modify the tkmvsenv_usermods.cfg file that is in the same directory.

Order of Precedence	Path and Filename	Purpose
3	z/OS: **/Lev1/**`server-context`**/ tkmvsenv.cfg**	Contains name/value pairs (environment variables) that apply to all components of the application server. Do not modify this file. If you need to make changes, modify the tkmvsenv_usermods.cfg file that is in the same directory.
4	z/OS: **/Lev1/**`server-context`**/ tkmvsenv_usermods.cfg**	Contains site-specific name/value pairs (environment variables) that apply to all application server components. Use this file to change the options that are specified in the tkmvsenv.cfg file in the same directory. See the *Configuration Guide: SAS Foundation for z/OS*.
5	z/OS: **/Lev1/**`server-context`**/**`server- name`**/tkmvsenv_usermods.cfg**	Contains site-specific modifications to the server's name/value pairs (environment variables). Use this file to change the options that are specified in the tkmvsenv.cfg file in the same directory. See the *Configuration Guide: SAS Foundation for z/OS*.

Configuration Files for SAS/SHARE Servers

The following table shows the configuration files that affect SAS/SHARE servers. For an explanation of each columns' contents, see "Introduction to the Configuration File Reference" on page 447.

Table 31.5 *Configuration Files for SAS/SHARE Servers*

Order of Precedence	Path and Filename	Purpose
1	Windows: **\Lev1\ShareServer\sasv9.cfg** UNIX and z/OS: **/Lev1/ShareServer/ sasv9.cfg**	Specifies start-up options for the server and contains calls to the other files that are listed in this table. Do not modify this file. If you need to make changes, modify the sasv9_usermods.cfg file that is in the same directory.
2	Windows: `SAS-install-directory` **\SASFoundation\9.4\sasv9.cfg** UNIX: `SAS-install-directory`**/ SASFoundation/9.4/sasv9.cfg** z/OS: Not applicable	Specifies system options that apply to all SAS sessions on the host machine. On UNIX systems, do not modify this file. If you need to make changes, modify the sasv9_local.cfg file that is in the same directory.
3	UNIX only: `SAS-install-directory`**/ SASFoundation/9.4/sasv9_local.cfg**	On UNIX systems, contains site-specific modifications to the sasv9.cfg file that is in the same directory.

Order of Precedence	Path and Filename	Purpose
4	Windows: *SAS-install-directory* **\SASFoundation\9.4***locale***\sasv9.cfg** UNIX: /SAS-install-directory/ SASFoundation/9.4/*locale*/sasv9.cfg z/OS: **<HLQ>.V93D8***ddd***.CNTLDSN(BAT***locale***W0)**	Specifies system options that apply to all SAS sessions on the host machine in the specified locale.
5	Windows: **\Lev1\ShareServer \sasv9_usermods.cfg** UNIX and z/OS: **/Lev1/ShareServer/ sasv9_usermods.cfg**	Contains site-specific modifications to the server's start-up options. Use this file to change the options that are specified in the sasv9.cfg file in the same directory. See the *SAS/SHARE User's Guide*.
6	Windows: **\Lev1\ShareServer \autoexec.sas** UNIX and z/OS: **/Lev1/ShareServer/ autoexec.sas**	Contains SAS statements that are executed immediately after SAS initializes on the server. Do not modify this file. If you need to make changes, modify the autoexec_usermods.sas file that is in the same directory.
7	Windows: **\Lev1\ShareServer \autoexec_usermods.sas** UNIX and z/OS: **/Lev1/ShareServer/ autoexec_usermods.sas**	Contains site-specific modifications to the autoexec.sas file that is located in the same directory.
Not applicable	Windows: \Lev1\sasv9_meta.cfg UNIX and z/OS: **/Lev1/sasv9_meta.cfg**	Specifies metadata server connection information for the SAS OLAP Server and SAS/SHARE server.
Not applicable	Windows: **\Lev1\ShareServer \logconfig.xml** UNIX and z/OS: **/Lev1//logconfig.xml**	Specifies the logging configuration for the server. See Chapter 10, "Administering Logging for SAS Servers," on page 131.
Not applicable	Windows: Not applicable UNIX and z/OS: **/Lev1/ShareServer.srv**	On UNIX and z/OS, contains information that is used by generate_boot_scripts.sh to regenerate the sas.servers script.

In z/OS environments, the following additional configuration files affect SAS/SHARE servers:

Table 31.6 *Additional Configuration Files for SAS/SHARE Servers*

Order of Precedence	Path and Filename	Purpose
1	z/OS: **<HLQ>.TKMVSENV(TKMVSENV)**	Contains name/value pairs (environment variables) that apply to all SAS sessions on the machine.

Order of Precedence	Path and Filename	Purpose
2	z/OS: **/Lev1/ShareServer/tkmvsenv.cfg**	Contains name/value pairs (environment variables) for the server. Do not modify this file. If you need to make changes, modify the tkmvsenv_usermods.cfg file that is in the same directory.
3	z/OS: **/Lev1/ShareServer/tkmvsenv_usermods.cfg**	Contains site-specific modifications to the server's name/value pairs (environment variables). Use this file to change the options that are specified in the tkmvsenv.cfg file in the same directory. See the *Configuration Guide: SAS Foundation for z/OS*.

Configuration Files for SAS Object Spawners and SAS/CONNECT Spawners

The following table shows the configuration files that affect SAS object spawners and SAS/CONNECT spawners. For an explanation of each columns' contents, see "Introduction to the Configuration File Reference" on page 447.

Table 31.7 *Configuration Files for the SAS Object Spawner*

Order of Precedence	Path and Filename	Purpose
Not applicable	Windows: **\Lev1*spawner-name*\metaConfig.xml** UNIX and z/OS: **/Lev1/*spawner-name*/metaConfig.xml**	Specifies metadata server connection information. The spawner must be stopped and restarted to pick up changes to this file.
Not applicable	Windows: **\Lev1\ConnectSpawner\ConnectSpawner.bat** UNIX: **/Lev1/ConnectSpawner/ConnectSpawner.sh** z/OS: Not applicable	Contains start-up options for the SAS/CONNECT spawner. See *SAS/CONNECT User's Guide*.
1	Windows: **\Lev1\ObjectSpawner**\ObjectSpawner.bat UNIX: **/Lev1/ObjectSpawner/ObjectSpawner.sh** z/OS: Not applicable	Contains start-up options for the object spawner. See "Spawner Invocation Options" in *SAS Intelligence Platform: Application Server Administration Guide*.
2	Windows: **\Lev1\ObjectSpawner**\ObjectSpawner_usermods.bat UNIX: **/Lev1/ObjectSpawner/ObjectSpawner_usermods.sh** z/OS: Not applicable	Can be used to specify site-specific modifications to the spawner's start-up options. Sites that need to set JREOPTIONS on the spawner can uncomment parameters that are included in this file.

Order of Precedence	Path and Filename	Purpose
Not applicable	Windows: `\Lev1\`*`spawner-name`*`\logconfig.xml` UNIX and z/OS: `/Lev1/`*`spawner-name/`*`logconfig.xml`	Specifies the logging configuration for the spawner. See Chapter 10, "Administering Logging for SAS Servers," on page 131.
Not applicable	Windows: `\Lev1\`*`spawner-name`*`\metadataConfig.xml` UNIX and z/OS: `/Lev1/`*`spawner-name/`*`metadataConfig.xml`	Provides information for the spawner to use when it connects to a clustered metadata server.
Not applicable	Windows: Not applicable UNIX and z/OS: `/Lev1/`*`server-or-`*`spawner-name.srv`	On UNIX and z/OS, contains information that is used by generate_boot_scripts.sh to regenerate the sas.servers script.

Reference Information for omaconfig.xml

About the omaconfig.xml File

The omaconfig.xml file specifies changes to standard features of the SAS Metadata Server, the repository manager, and policies related to internal users. The features that can be changed include the following:

- the names and locations of the adminUsers.txt and trustedUsers.txt files

- the number of threads that are allowed to run concurrently on the SAS Metadata Server (contingent on the number of available processors)

- the email addresses to which alert emails are to be sent

- journaling options

- the location, the libref, and the engine used by the repository manager

- policies regarding password assignment and management for internal users

- option to perform an analysis of metadata repository files to determine damage or to repair a damaged repository upon server start

For options that are omitted from the omaconfig.xml file, the server uses the option's default value.

Your omaconfig.xml file might include an option that activates a SAS Metadata Server or repository manager feature that is needed by a particular SAS application. When this is the case, do not remove the option or change its setting unless you are instructed to do so by SAS Technical Support.

The omaconfig.xml file is located in the following path: *`SAS-configuration-directory/Lev1/SASMeta/MetadataServer`*.

When you change any option in omaconfig.xml, you must stop and restart the metadata server for the change to take effect. For instructions, see Chapter 6, "Operating Your Servers," on page 61.

> *CAUTION:*
> **Stopping the metadata server can stop other components.**

Repair and Analyze Metadata Files Options

The omaconfig.xml file can contain the following XML elements that contain debugging options that provide the same functionality as the Metadata Analysis and Repair tools on page 236 in SAS Management Console. These XML elements can be used to analyze and repair the repository files upon the initial start-up of the metadata server process for a server that is failing to start. It is not recommended that you leave these options in once a metadata server starts successfully, because doing so can significantly affect metadata server performance. Observe the metadata server log for errors regarding the analysis or repair.

<DEBUG ANALYZE_METADATA_FILES="y"/>
   performs analysis of SAS Metadata Repository files for damage.

   Example of usage in omaconfig.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<OMAconfig>
   <DEBUG ANALYZE_METADATA_FILES="y"/>
```

<DEBUG REPAIR_METADATA_FILES="y"/>
   performs analysis of SAS Metadata Repository files for damage and immediately repairs many types of corruption in the SAS Metadata Repository.

   Example of usage in omaconfig.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<OMAconfig>
   <DEBUG REPAIR_METADATA_FILES="y"/>
```

<OMA>
   specifies options for the metadata server. See "<OMA> Options" on page 457.

<RPOSMGR>
   specifies options for the repository manager.

<InternalAuthenticationPolicy>
   specifies options for the assignment and management of passwords for internal users. Internal accounts are accounts that exist only in metadata. See "SAS Internal Authentication" in *SAS Intelligence Platform: Security Administration Guide*.

The following topics provide details for each omaconfig.xml option.

<OMA> Options

The <OMA> element specifies the following options for the metadata server:

ADMINUSERS="*path-and-filename*"
   specifies the name and location of the adminUsers.txt file. A planned installation creates the file in **SAS-configuration-directory/Lev1/SASMeta/ MetadataServer/adminUsers.txt**.

ALERT_CONDITION_FREQUENCY="*time*"
   specifies the amount of time between reminder emails about metadata server alert conditions. Use any of the following formats to specify the amount of time:

   • *number-of-seconds*s (for example, **"51600s"**) or just *number-of-seconds* (for example, "51600")

- *number-of-minutes*m (for example, **"180m"**)

- *number-of-hours*h (for example, **"48h"**)

- *number-of-days*d (for example, **"3d"**)

This option is new with SAS 9.4M2.

> **Default**   6h

ALERT_CONDITION_GRACE_PERIOD="*time*"
specifies the amount of time an alert condition persists before the metadata server shuts itself down. To specify the time, use the same formats that are allowed for ALERT_CONDITION_FREQUENCY.

This option is new with SAS 9.4M2.

> **Default**   3d

ALERTEMAIL="*email-address*"
specifies an email address to which the metadata server sends a notification message in the event of a metadata server backup error, a metadata server recovery error, or an error that prevents the repository data sets from being updated from the journal. For clustered metadata servers, a message is also sent if one of the worker nodes fails to initialize or synchronize and shuts itself down.

To specify more than one address, enclose each address in single quotation marks, place a blank space between each address, and enclose the list in parentheses. For example: **"('Bill@mycompany.com' 'Susan@mycompany.com')"**

> **Interaction**   For the ALERTEMAIL and ALERTEMAILTEST options to work, the appropriate mail server connection properties (EMAILSYS, EMAILHOST, EMAILPORT, and EMAILID) must be specified in the metadata server's sasv9.cfg or sasv9_usermods.cfg file. If your email server requires authentication, the appropriate information must be provided in EMAILAUTHPROTOCOL, EMAILID and EMAILPASSWORD. See Chapter 32, "Options for the Metadata Server Invocation Command," on page 469.

> **Tip**   If you modify any of the email options, be sure to test metadata server email alerts. See "Testing Email Alerts for the SAS Metadata Server" on page 205.

> **See**   "Managing Alert Email Options for the SAS Metadata Server" on page 204 for more information.

EMAILSSL="NORMAL|SSL|STARTTLS"
specifies to use the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocol for communications between the SMTP email server and the metadata server. The SMTP email server is identified in the EMAILHOST= system option.

You can specify one of the following protocols:

NORMAL   specifies to use a non-secured communication. This is the default value if neither EMAILHOST= or EMAILSSL= specifies SSL or STARTTLS.

SSL   specifies that the connection is secured from the start.

STARTTLS   specifies to take an unsecured connection and make it secure.

Note: If the EMAILHOST= system option specifies the SSL or STARTTLS option, you do not need to set the EMAILSSL= omaconfig.xml option. When both EMAILHOST= and EMAILSSL= specify a protocol and the protocols are different, the EMAILHOST= setting takes precedence. However, if EMAILSSL= specifies a protocol and EMAILHOST= does not, then the EMAILSSL= protocol is used.

ALERTEMAILTEST="*text*"

enables you to verify that the ALERTEMAIL option is set correctly. ALERTEMAILTEST is a temporary omaconfig.xml option that causes a test message to be sent at server start up in order to verify that the email addresses specified in the ALERTEMAIL option are valid. An error causes the email not to be sent and the metadata server to be stopped, so that the administrator can investigate which option has a problem. The metadata server is also stopped when the email is sent correctly so that the administrator can remove the ALERTEMAILTEST option from the omaconfig.xml file. The SAS Metadata Server will not run with the ALERTEMAILTEST option set. Be sure to specify a text string with this option. Using the option without a text string can also prevent the email from being sent.

Note The **Send Test Message** button in SAS Management Console is the preferred method for testing alert emails. This feature enables you to test alert emails without stopping the metadata server. The button is located on the Active Server Properties dialog box in the Metadata Manager plug-in. See "Testing Email Alerts for the SAS Metadata Server" on page 205.

GROUPSFORMEMBERASSOCDELETE="*comma-separated-list-of-groups*"

invokes special functionality that is needed by SAS Information Delivery Portal. This option specifies the names of groups that allow a member to be deleted by a user who has WriteMetadata permission for the member, even though the user does not have WriteMetadata permission for the group itself. Only the groups listed on this option allow the behavior; other groups continue to use the default security for the membership list. The following groups need special processing:

- DESKTOP_PORTALPAGES_GROUP

- Portlet Template Group

- OrderingGroup

- DESKTOP_PAGEHISTORY_GROUP

- Portal Collection

The names are case sensitive. The GROUPSFORMEMBERASSOCDELETE option is added to the omaconfig.xml file by the planned SAS installation. Do not change the names unless instructed to do so by SAS Technical Support. If this option is removed from the omaconfig.xml file, owners of information maps, reports, and stored processes, as well as others who have WriteMetadata permission to the metadata describing these items, will not be able to delete the items if someone has created a bookmark to them in the SAS Information Delivery Portal.

JOURNALPATH="*path-and-filename*"

specifies a location to create the journal file if JOURNALTYPE="SINGLE" has been specified. (If JOURNALTYPE is set to NONE or ROLL_FORWARD, then the JOURNALPATH option is ignored.)

Note: The settings JOURNALTYPE="SINGLE" and JOURNALTYPE="NONE" are not recommended.

When specifying a location, keep these things in mind:

- For consistency, we recommend that you specify the filename
  MetadataServerJournal.dat.

- For convenience and ease of management, the standard metadata server configuration creates repositories and the metadata server log on the same file system in which the metadata server is started. If it is at all likely that this file system fills up, consider moving the repositories to another file system. If the server file system fills up, the journal file is not threatened because it is a fixed size. However, the metadata server log needs space to be able to record any disk-full messages in order to send them to the metadata server administrator.

- Put the journal file on a native file system, such as the C: (system) drive. The location should be fast and highly reliable. Do not place the file on a network-mounted file system or network appliance.

- If you use the same file system in which the metadata server is started, use a relative pathname to identify the file's location. The pathname should be relative to the directory from which the metadata server is started.

Default	If you specify JOURNALTYPE="SINGLE" but you do not specify a JOURNALPATH, then the journal is created in *SAS-configuration-directory*/Lev1/SASMeta/MetadataServer/Journal/MetadataServerJournal.dat.
Interaction	This option is ignored if OMA JOURNALTYPE is set to NONE or ROLL_FORWARD.

JOURNALSIZE="*number-of-bytes*"
specifies a size, in bytes, for the journal file if JOURNALTYPE="SINGLE" has been specified. (If JOURNALTYPE is set to NONE or ROLL_FORWARD, then the JOURNALSIZE option is ignored.)

Note: The settings JOURNALTYPE="SINGLE" and JOURNALTYPE="NONE" are not recommended.

The space that you assign is reserved in the file system and is unavailable to other processes, whether the journal file is full or not. Journal entries are appended to the file until the file grows to the size that is specified. When the file reaches the specified size, subsequent journal entries overwrite previously written entries starting at the beginning of the file.

If clients regularly experience delays in getting their add, update, and delete requests processed, check the SAS Metadata Server log. If you find messages similar to the following, you might consider setting a higher JOURNALSIZE value: **WARNING: insufficient file space available in journal file.**

Default	If you specify JOURNALTYPE="SINGLE" but you do not specify a JOURNALSIZE, then the default size is used, as follows: 200MB on 32–bit machines and z/OS machines, and 500MB on 64–bit machines.
Interaction	This option is ignored if OMA JOURNALTYPE is set to NONE or ROLL_FORWARD.

JOURNALTYPE="NONE | SINGLE | ROLL_FORWARD"
specifies whether metadata server journaling is enabled and, if so, which type of journal file is created. When journaling is enabled, access is returned to clients as soon as the metadata updates are written to the in-memory database and the journal file. The more time-consuming updates to the repository data sets are performed later in the background. If the metadata server fails before the update process has applied

all of the updates from the journal file, the server recovers the updates from the journal file when it is restarted.

You can specify one of the following journal types:

NONE	(Not recommended) disables the journaling feature so that no journal file is created. The JOURNALPATH and JOURNALSIZE options are ignored. The roll-forward option is not available when the metadata server is recovered from a backup. In addition, the metadata server cannot be backed up while the server is in an ONLINE state. The server pauses itself to a READONLY state each time a backup is executed, and any update transactions that are issued during the backup will fail.
SINGLE	(Not recommended) enables the journaling feature by creating a single, circular journal file of a fixed size at the location that is specified in JOURNALPATH. Journal entries are appended to the file until the file grows to the size that is specified in JOURNALSIZE. When the file reaches the JOURNALSIZE limit, subsequent journal entries overwrite previously written entries starting at the beginning of the file. The roll-forward recovery option is not supported because there is no guarantee that all transactions that occurred since the last backup will still be available when a recovery is executed.
ROLL_FORWARD	enables the journaling feature by creating a linear journal file that permanently stores all transactions that have occurred since the most recent backup. The journal file is written to the same location as the associated backup files. Each time a new backup is executed, journaling stops and a new journal file is started in the new backup location. The JOURNALPATH and JOURNALSIZE options are ignored. If ROLL_FORWARD is specified, the roll-forward option is available when the metadata server is recovered from a backup. If you change the option setting to ROLL_FORWARD from NONE or SINGLE, a metadata server backup is automatically performed when the server is restarted.

Default	The SAS Deployment Wizard sets this option to ROLL_FORWARD during installation or during migration from SAS 9.2.
Interaction	If ROLL_FORWARD or NONE is specified, then JOURNALPATH and JOURNALSIZE are ignored.
Tip	For optimum performance and to ensure system integrity in the event of a failure, it is strongly recommended that journaling be enabled at all times.
See	"Configuring Metadata Server Journaling" on page 202
CAUTION	**The ROLL_FORWARD option must be specified if you are using metadata server clustering. It is also required if you want to be able to use the roll-forward option in the event of a metadata server recovery.**

JOURNALYIELDCOUNT="*number-of-observations***"**
    is designed to prevent large repository updates (for example, requests that add
    hundreds of objects) from monopolizing the server. It specifies the number of records
    that the update process can apply before yielding to allow other internal processing
    tasks, such as writing to the metadata server log, to occur and then resuming
    operation. For example, if JOURNALYIELDCOUNT were set to 20 and a work unit
    consisted of 100 records, the update process would apply 20 records, stop to allow
    internal processes to function, apply 20 more records, and so on, until all 100 records
    were applied to repositories.

    This option does not affect the order in which updates are applied to repositories.
    Metadata updates are applied in the order in which they are received, and all
    repositories are locked for reading and writing at proper times.
    JOURNALYIELDCOUNT improves the overall performance of the metadata server
    by allowing those updates that have been performed on the in-memory metadata
    copy to be applied to the permanent copy on disk during periods of CPU availability.
    The in-memory copy remains up-to-date at all times.

    The maximum setting of the JOURNALYIELDCOUNT option is 100 records; the
    minimum setting is 5 records. The default value is 14 records. Do not adjust the
    default value unless instructed to do so by SAS Technical Support.

MAXACTIVETHREADS="*number-of-threads***"**
    specifies the number of threads that are allowed to run concurrently on the metadata
    server. The number of processors determines the number of concurrent queries that
    can be made. If the server has only one processor, then the recommended setting is 2.
    If the server has more than one processor, then the recommended setting is the
    number of processors. A setting that is higher or lower than the recommended one
    can affect performance.

    For information about how the metadata server uses threads, and recommendations
    for setting the MAXACTIVETHREADS option and the related THREADSMIN and
    THREADSMAX server invocation options, see "Configuring the Number of
    Threads Used by the Metadata Server" on page 200.

    Default   If the host machine has one processor, then the maximum active number of
                threads is set to 2. If the host machine has two or more processors, then the
                maximum active number of threads is set to the number of processors.

MAXIMUM_QUERY_RETRY_COUNT="*number-of-retries***"**
    specifies the maximum number of times that a metadata query request involving
    multiple repositories can be restarted because of missing repository locks. If the
    maximum number of retries (also referred to as the "query retry limit") is reached
    before the query request is completed, a repository lock error is returned.

    When the metadata server receives a query request, the server does the following to
    process the request:

    1. obtains a shared query lock for each repository that is specified in the request.

    2. as the request is prepared, verifies that a shared query lock has been obtained for
        each repository that contains records that are needed to complete the request.

    3. immediately attempts to obtain any shared query locks that were not obtained
        initially.

    4. restarts the request if the missing locks are not available. Each restart constitutes
        a retry.

5. repeats the process until the request is complete or until the query retry limit is reached. If the query request cannot be completed within the query retry limit, a repository lock error is returned.

Activity that is related to obtaining repository query locks is recorded in the Metadata.CounterQueryUpdateLock counters. See "Using the Server Performance Counters and Information Fields" on page 121.

Default 7 or the number of repositories that were configured when the metadata server was last started, whichever is higher

MAXIMUM_RETRY_COUNT="*number-of-retries*"

specifies the maximum number of times that a metadata update request involving multiple repositories can be restarted because of missing repository locks. If the maximum number of retries (also referred to as the "update retry limit") is reached before the update request is completed, a repository lock error is returned.

When the metadata server receives an update request, the server does the following to process the request:

1. obtains an exclusive update lock for each repository that is specified in the request.

2. prepares the request, building a transaction of all the attributes, associations, and associated objects that need to be updated, added, or deleted in any repository to successfully complete the request.

3. before committing the transaction, verifies that exclusive update locks have been obtained for all repositories that are being updated, and restarts the request if any update locks are missing.

4. repeats the process until the request is complete or until the update retry limit is reached. If the update request cannot be completed within the retry limit, a repository lock error is returned.

Activity that is related to obtaining repository update locks is recorded in the Metadata.CounterRetryUpdateLock counters. See "Using the Server Performance Counters and Information Fields" on page 121.

Default 3

SASSEC_LOCAL_PW_SAVE="1 | Y | T | 0 | N | F"

specifies whether users of desktop applications can save their user IDs and passwords in a local metadata connection profile.

If you specify `1`, `Y`, or `T`, then a **Save user ID and password in this profile** check box is available to users when they launch desktop applications. If a user selects this check box, then the user's ID and password are stored in the user's connection profile on the local file system. This profile is used for future connections to the metadata server. Saving the ID and password in the profile allows users to reconnect in the future without entering these values again.

If you do not want to allow IDs and passwords to be stored in client connection profiles, specify `0`, `N`, or `F`.

The initial setting for this parameter is 1.

Notes After you change the setting for this option and restart the metadata server, each client uses the previous setting for its first connection. The client then

discovers the revised setting and conforms to that revised setting for subsequent connections.

If you change the setting to disallow saved credentials, and credentials are already present in a user's connection profile, those credentials must be manually removed.

For a few solutions desktop clients (for example, SAS Model Manager, SAS Enterprise Miner, and SAS Forecast Studio), the ability to store credentials in client-side connection profiles is instead controlled by the Policy.AllowClientPasswordStorage property. To access this property, open the **Plug-ins** tab of SAS Management Console and navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **Settings** ⇨ **Policies** ⇨ **Allow client password storage**.

RETURNPASSWORDS="CLEAR|SAS001|SAS002|SAS003|SAS004|SAS005"
   causes passwords to be encoded when they are returned from the metadata server to a requesting client.

   Important: Starting in SAS 9.4M8, the prerequisite for SAS003 and higher is having the required cryptographic libraries in the OS. In earlier releases, the prerequisite for SAS003 and higher is SAS/SECURE. If an applicable prerequisite is not met, SAS002 is used (regardless of the value of the RETURNPASSWORDS option).

   You can specify one of the following encoding methods:

   CLEAR   specifies no encoding of passwords.

   SAS001   specifies the use of base64 to encode passwords.

   SAS002   specifies the SASProprietary encoding method. This method is included in SAS software and uses a 32–bit key to encode passwords.

   SAS003   specifies the Advanced Encryption Standard (AES) encoding method with a salt size of 16 bits.

   SAS004   specifies the AES encoding method with a salt size of 64 bits.

   SAS005   specifies the AES encoding method with a salt size of 64 bits and 10000 iterations for password storage.

   The initial setting for this parameter is SAS002.

   Default   The default setting is SAS002 or, in a SAS 9.4M7 or earlier deployment that has SAS/SECURE and does not specify a value for RETURNPASSWORDS, SAS003.

STOREPASSWORDS="SAS002|SAS003|SAS004|SAS005"
   causes passwords to be encoded when they are stored in metadata.

   Important: Starting in SAS 9.4M8, the prerequisite for SAS003 and higher is having the required cryptographic libraries in the OS. In earlier releases, the prerequisite for SAS003 and higher is SAS/SECURE. If an applicable prerequisite is not met, SAS002 is used (regardless of the value of the STOREPASSWORDS option).

   You can specify one of the following encoding methods:

   SAS002   specifies the SASProprietary encoding method. This method is included in SAS software and uses a 32–bit key to encode passwords.

SAS003   specifies the Advanced Encryption Standard (AES) encoding method with a salt size of 16 bits.

SAS004   specifies the AES encoding method with a salt size of 64 bits.

SAS005   specifies the AES encoding method with a salt size of 64 bits and 10000 iterations for password storage.

Default   The default setting is SAS002 or, in a SAS 9.4M7 or earlier deployment that has SAS/SECURE and does not specify a value for STOREPASSWORDS, SAS003.

CAUTION   **If you discontinue conformance with an applicable prerequisite, stored passwords are unusable and inaccessible.** Do not uninstall SAS/SECURE or required cryptographic libraries without first converting all stored passwords to SAS002 format. See "How to Re-encrypt Stored Passwords" in *SAS Intelligence Platform: Security Administration Guide*.

TRUSTEDUSERS="*path-and-filename*"
specifies the name and location of the trustedUsers.txt file. The planned installation creates the file in **SAS-configuration-directory/Lev1/SASMeta \MetadataServer**

<RPOSMGR> Options

The <RPOSMGR> element specifies the following options for the repository manager:

LIBREF="*name*"
specifies an alternate libref for the repository manager. *Name* must be a valid SAS name. The name can be up to eight characters. The first character must be a letter (A, B, C, . . ., Z) or an underscore (_). Subsequent characters can be letters, numeric digits (0, 1, . . ., 9), or underscores. You can use uppercase or lowercase letters. SAS processes names as uppercase, regardless of how you enter them.

Default   RPOSMGR

ENGINE="*libname-engine*"
specifies the LIBNAME engine that is to be used to create the repository manager. The default value is an empty string; it defaults to Base SAS. When changing the default value, note that engine values must be specified in uppercase letters.

PATH="*pathname*"
specifies an alternate location for the repository manager. The planned installation creates the repository manager in **SAS-configuration-directory/Lev1/ SASMeta/MetadataServer/rposmgr**. *Pathname* can be an absolute reference or a reference relative to the metadata server start-up directory.

OPTIONS="*libname-options*"
specifies LIBNAME options for the repository manager. You can specify any option that is valid in a LIBNAME statement (for example, **BLKSZ=""**).

<InternalAuthenticationPolicy> Options

The <InternalAuthenticationPolicy> element specifies the following options for the assignment and management of passwords for internal users. Internal accounts are

accounts that exist only in metadata. See "SAS Internal Authentication" in *SAS Intelligence Platform: Security Administration Guide*.

CAUTION:

The option names for <InternalAuthenticationPolicy> are case sensitive and must be entered exactly as they are shown in this list.

ChangeDelayInMinutes="*number-of-minutes*"

specifies the number of minutes that must elapse following a password change before an internal user's password can be changed again.

Note: This option does not apply to administrators who are setting passwords for other users.

Default	0
Range	0–1440

DigitRequired="1 | Y | T | 0 | N | F"

specifies whether passwords for internal users must include at least one digit. To enforce this requirement, specify **1**, **Y**, or **T**. To disable this requirement, specify **0**, **N**, or **F**.

Default	F

ExpirationDays="*number-of-days*"

specifies the number of days after an internal user's password is created or changed that the password will expire and need to be changed. A value of 0 specifies that internal user's passwords will never expire.

Default	0
Range	0–32767
Note	You can use SAS Management Console to specify a different number of expiration days for a particular internal user's password.

ExpirePasswordOnReset="1 | Y | T | 0 | N | F"

specifies whether passwords for internal accounts are to be considered already expired when the account is created, so that users are required to change their passwords the first time they log on. To enforce this requirement, specify **1**, **Y**, or **T**. To disable this requirement, specify **0**, **N**, or **F**.

This option does not apply to administrative users who reset their own passwords in SAS Management Console. When this option is in effect, administrative users who reset their own passwords in SAS Management Console are not required to change their passwords again the next time they log on.

You can waive the password change requirement for a specific internal user. To do so, select the **never expire** option in SAS Management Console when you create the user's account or when you reset the user's password. Then a password change will not be required the first time the user logs in.

Note: The **never expire** option is automatically set for the initial internal users that are created by the SAS Deployment Wizard.

Default	T
Interaction	This option is valid only if Expiration Days is greater than 0.

HashPasswords="SHA256 | SHA256–10000 | MD5"
specifies the hash function to use on the internal account password that is stored in the metadata.

Important: Starting in SAS 9.4M8, the prerequisite for SHA–256 and higher is having the required cryptographic libraries in the OS. In earlier releases, the prerequisite for SHA–256 and higher is SAS/SECURE. If an applicable cryptographic library is not installed, MD5 is used.

Note:

- The SHA–256 (secure hash algorithm) hash function is FIPS (Federal Information Processing Standard) compliant.

- The MD5 (message digest algorithm 5) hash function is appropriate for preventing accidental exposure of information.

Default   SHA256–10000

MinLength="*number-of-bytes*"
specifies the minimum allowed length for the passwords of internal users.

Default   6

Range   1–32

MixedCase="1 | Y | T | 0 | N | F"
specifies whether passwords for internal users must include at least one uppercase letter and at least one lowercase letter. To enforce this requirement, specify `1`, `Y`, or `T`. To disable this requirement, specify `0`, `N`, or `F`.

Default   F

NumPriorPasswords="*number-of-previous-passwords*"
specifies the number of previous passwords that are to be stored in each internal user's password history.

Default   5

Range   0–5

InactiveDaysToSuspension="*number-of-days*"
specifies the number of days after which an internal user's account will be suspended if the account is not used. If you specify 0, then accounts can be inactive indefinitely without being suspended.

Note: Beginning in SAS 9.4M4, this option is no longer supported.

Default   0

Range   0–32767

LockoutDurationInMinutes="*number-of-minutes*"
specifies the number of minutes that an internal user must wait before trying to log on again after having been locked out because of excessive login failures. (See "NumFailuresForLockout="*number-of-failures*"" on page 468.)

Default   60

Range $1–2^{31}$

NumFailuresForLockout="*number-of-failures***"**

specifies the number of authentication failures after which an internal user's account will be locked out.

CAUTION:

If you specify 0, your system could become vulnerable to password guessing attacks.

Default   3

Range   0–100

Chapter 32

Options for the Metadata Server Invocation Command

Modifying the Metadata Server Invocation Options

The invocation options for the metadata server are specified in the sasv9.cfg file that is located in the following path:

***SAS-configuration-directory*/Lev1/*server-context*/MetadataServer**

During installation, this file is populated with the options that are appropriate for your site. Some of the values are based on information that the installer entered in response to SAS Deployment Wizard prompts.

If you want to specify different values for these options, or if you want to specify additional options, then enter your updates and additions in sasv9_usermods.cfg, which is located in the same directory as sasv9.cfg. You must restart the server in order for the changes to take effect.

Note: In configuration files on z/OS, do not precede system options with a hyphen.

For information about the options, see:

- "Recommended Parameters" on page 470
- "Optional Parameters" on page 473

Recommended Parameters

Recommended Parameters for All Host Environments

The following parameters are recommended for the optimal operation of the SAS Metadata Server in all host environments. Not all of the parameters are required to start the server. However, they are included here because they are considered helpful to the operation of the server.

- the pathname of the configuration file for SAS Foundation on the host machine.

- the name of the file that contains user modifications to the invocation options.

- the environment control parameters NOAUTOEXEC, NOTERMINAL, SASUSER, and RSASUSER:

  NOAUTOEXEC
  > specifies not to process the SAS autoexec file, even if one exists. Processing of the file can result in unpredictable server behavior.

  NOTERMINAL
  > specifies to run the server in batch mode.

  SASUSER="*library-specification*"
  > specifies the SAS library that is to contain the server's Profile catalog. This library must be dedicated to the metadata server and is typically a directory named `sasusrms` or `sasuser` under the `MetadataServer` directory.

  RSASUSER
  > limits access to the SASUSER library to Read-Only mode.

- the following parameters which are used for generating email alerts for the metadata server:

  EMAILHOST *server-network-address*
  > specifies the network address of the enterprise's SMTP server (for example, `mailhost.company.com`). Beginning in the third maintenance release of SAS 9.4, you can specify use of the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocol for communication between the SMTP email server and the metadata server. To configure use of SSL or TLS, specify the appropriate option in the EMAILHOST= system option or specify the EMAILSSL= option in the omaconfig.xml file. For more information about EMAILHOST= and EMAILSSL=, see "Reference Information for omaconfig.xml" on page 456. For more information about SSL and TLS, see "EMAILHOST=" in *SAS System Options: Reference*.

  EMAILPORT *port-number*
  > specifies the port number that is used by the SMPT server that you specified for the EMAILHOST option.

  EMAILID "*server-email-address | server-user-id*"
  > specifies the email address that is to appear in the **From** field for alert email messages. By default, the SAS Deployment Wizard places the following value in this parameter: `SASMeta - Metadata Server (Config - Lev1) <noreply@mailhost.fyi.sas.com>`.
  >
  > If you want to specify a different email address, use one of the following formats:

- *"server-name<user-account@domain>"*

- *"<user-account@domain>"*

  Note:

  - Some electronic mail systems do not allow semicolons or colons in the **From** field.

  - If your email server requires user authentication, then this option specifies the user ID that is to be used for authentication when an email alert is generated. See "Optional Parameters" on page 473.

  T I P   If you modify any of the email options, be sure to test metadata server email alerts. See "Testing Email Alerts for the SAS Metadata Server" on page 205.

  Note:

  - If your mail server requires authentication, then additional parameters are needed. See "Optional Parameters" on page 473.

  - For more information about metadata server alerts, see "Managing Alert Email Options for the SAS Metadata Server" on page 204.

- the logging-related parameters LOGCONFIGLOC, PAGESIZE, and LINESIZE:

  LOGCONFIGLOC=*file-specification*
    specifies the name of the XML configuration file that is used to initialize the SAS logging facility for the server. *file-specification* must be a valid filename or a path and filename for your operating environment. If the path contains spaces, enclose *file-specification* in quotation marks.

  PAGESIZE=*value*
    specifies the number of lines that make up a page of SAS output. A setting of **MAX** is recommended to reduce the occurrence of SAS log page headings and carriage controls.

  LINESIZE=*value*
    specifies the line size of SAS output. A setting of **MAX** is recommended to reduce log entry wrap-around.

- the security-related parameters NETENCRYPTALGORITHM and SSPI:

  NETENCRYPTALGORITHM="*algorithm1*" | ("*algorithm1*", "*algorithm2*", ...)
    set in conjunction with the CLIENTENCRYPTIONLEVEL OBJECTSERVERPARMS option, specifies the algorithm(s) to be used for encrypted client/server data transfers. If you specify more than one algorithm, enclose all the algorithm names in parentheses and use commas to separate them. If there are embedded blanks in an algorithm name, enclose each algorithm within quotation marks. Valid values are **SASPROPRIETARY**, **RC2**, **RC4**, **DES**, and **TRIPLEDES**. NETENCRALG is an alias for NETENCRYPTALGORITHM.

  SSPI
    specifies support for the Security Support Provider Interface for single sign on (SSO) connections to IOM servers.

- OBJECTSERVER, to invoke an Integrated Object Model (IOM) server.

- OBJECTSERVERPARMS, to pass IOM parameters to the server:

  CLIENTENCRYPTIONLEVEL=NONE | CREDENTIALS | EVERYTHING
    set in conjunction with the NETENCRYPTALGORITHM option, this option specifies the degree of encryption that the server uses when making outbound calls.

NONE
> specifies that nothing is encrypted. Clients transmit user IDs and passwords as plain text.

CREDENTIALS
> specifies that clients encrypt user IDs and passwords.

EVERYTHING
> specifies that clients encrypt all client/server communications.

> CEL is an alias for CLIENTENCRYPTIONLEVEL.

PROTOCOL=*value*
> identifies the network protocol that is used to communicate with the server. The only valid value is **bridge**.

PORT=*port-number*
> specifies the TCP port on which the metadata server will listen for requests and that clients will use to connect to the server. The *port-number* value must be a unique number between 0 and 64K.

CLASSFACTORY=*factory-number*
> specifies the type of IOM server to instantiate. The value **0217E202-B560-11DB-AD91-001083FF6836** identifies a SAS Metadata Server.

TRUSTSASPEER=*pathname*
> enables SAS peer sessions from IOM servers to connect as trusted peer sessions. If you specify a blank or empty file, any SAS peer session can connect as a trusted peer. If you specify a file that contains a list of trusted domains, SAS peer sessions are trusted only if they belong to a domain that is listed in your trusted peer file.

Note:

- For details about the OBJECTSERVER option, the SSPI option, and the OBJECTSERVERPARMS option and its parameters, see "System Options for SAS Application Server Components" in *SAS Intelligence Platform: Application Server Administration Guide*.

- If you need to make any changes to the OBJECTSERVERPARMS option, make sure that the sasv9_usermods.cfg file contains the entire list of parameters that you want to specify for this option. (The list of object server parameters that you specify in sasv9_usermods.cfg overrides the list that is specified in sasv9.cfg.

- For details about the other options, see the *SAS System Options: Reference*.

Additional Windows Parameters

A Windows server also requires the following parameters:

The system administration parameter MEMSIZE=*value*
> specifies the amount of memory to allocate to the metadata server process. We recommend a value that allows the operating system to set the memory limits on the server. This can be accomplished by setting a value of **0** (zero), **MIN**, or **MAX**.

The display parameter NOSPLASH
> suppresses the display of SAS logo and copyright information when invoking the SAS session.

SASINITIALFOLDER
> specifies the path to the server's working folder.

For more information, see the *SAS Companion for Windows*.

Additional UNIX Parameters

A UNIX server requires the following parameter in addition to those described in "Recommended Parameters for All Host Environments" on page 470:

The system administration parameter MEMSIZE=*value*
> specifies the amount of memory to allocate to the metadata server process. We recommend a value that allows the operating system to set the memory limits on the server. This can be accomplished by setting a value of **0** (zero), **MIN**, or **MAX**.

For more information, see the *SAS Companion for UNIX Environments*.

Additional z/OS Parameters

A z/OS server requires the following parameters in addition to those described in "Recommended Parameters for All Host Environments" on page 470:

The file control parameter FILESYSTEM=HFS
> specifies to run the server in a hierarchical file system.

The OBJECTSERVERPARMS parameter THREADSSTACKSIZE=1048576
> specifies the stack size to give each thread.

For more information, see the *SAS Companion for z/OS*.

Optional Parameters

The following parameters invoke optional functionality in a SAS Metadata Server in all host environments:

The following parameters, which might be needed to generate email alerts for the metadata server:

EMAILID "*server-email-address | server-user-id*"
> specifies the user ID that is to be used for authentication when an email alert is generated. Use this option for this purpose only if your email server requires user authentication.

EMAILPW "*server-email-password*"
> specifies the password that is associated with the user ID that is specified in EMAILID. Specify this option only if your email server requires user authentication. The password should be encrypted using SAS proprietary 32-bit encryption. To obtain the encrypted password, use PROC PWENCODE.
>
> *Note:* If you provide this password in the SAS Deployment Wizard, then the wizard automatically encrypts it. To update this password, you can use the Update Passwords feature of the SAS Deployment Manager. See "About Updating Managed Passwords" in *SAS Intelligence Platform: Security Administration Guide*.

EMAILAUTHPROTOCOL LOGIN | NONE
> specifies the protocol that is to be used to authenticate the user to the email server. If your email server requires user authentication, specify a value of LOGIN. Otherwise, omit this option or specify the default value (NONE).

> *TIP* If you modify any of the email options, be sure to test metadata server email alerts. See "Testing Email Alerts for the SAS Metadata Server" on page 205.

Note: For more information about metadata server alerts, see "Managing Alert Email Options for the SAS Metadata Server" on page 204.

AUTHPROVIDERDOMAIN=(*provider*:*domain-name*)
  specifies an alternative authentication provider. When the AUTHPROVIDERDOMAIN parameter is omitted from the metadata server start command, the metadata server uses host authentication to authenticate users. AUTHPD is an alias for AUTHPROVIDERDOMAIN. For usage information, see "Direct LDAP Authentication" in *SAS Intelligence Platform: Security Administration Guide*.

  Note: In UNIX operating environments, you must insert an escape character before each parenthesis. For example:`-authproviderdomain = \ (ADIR:MyDomain\)`

The OBJECTSERVERPARMS parameters:

THREADSMIN=*minimum-number-of-threads*
  specifies the minimum number of threads in the server's thread pool. When the number of threads falls below this number, IOM mechanisms automatically create new threads. TMIN is an alias for THREADSMIN. THREADSMIN and THREADSMAX are set in conjunction with the MAXACTIVETHREADS server configuration option.

  By default, the server uses this formula to compute the minimum number of threads:

  `MAX(5,((number-of-processors * 2) + 1))`

  This means that the value is set to either 5 or to ((*number-of-processors* * 2) + 1), whichever is greater.

THREADSMAX=*maximum-number-of-threads*
  specifies the maximum number of threads in the server's thread pool. When the number of threads exceeds this number, IOM mechanisms automatically delete threads. TMAX is an alias for THREADSMAX.

  Note: It is recommended that TMIN=TMAX.

  THREADSMIN and THREADSMAX are set in conjunction with the MAXACTIVETHREADS server configuration option.

  Default By default, if no value is specified for maximum number of threads, it is set to the computed value of THREADSMIN.

Note:

- For details about the OBJECTSERVERPARMS option and its parameters, see the *SAS Intelligence Platform: Application Server Administration Guide*.

- For details about the AUTHPROVIDERDOMAIN option, see the *SAS System Options: Reference*.

Using the SAS Deployment Manager to Update Host Name References

Overview of Updating Host Name References

About the Update Host Name References Tool in SAS Deployment Manager

The SAS Deployment Manager includes an option called Update Host Name References. This option, which is referred to here as the Update Host Name References tool, provides an automated way to update references to server host names in your SAS deployment. You can use this tool if the host name or the network domain name of an existing server machine has changed. You can also use it if you have created an exact clone of one or more servers (or an entire deployment) on another machine or machines.

To use the tool, you launch the SAS Deployment Manager on each machine in your deployment and choose the Update Host Name References option. For each existing machine whose name has changed (or each machine that you have cloned to a new machine), you specify the old host name and the new host name. You can specify all forms of the old and new name, including the fully qualified name, the short name, and the IP address. Or, if you are cloning a machine, you can configure the tool to access the Domain Name System to determine the various forms of the old and new names.

The tool then searches metadata, configuration files, and other files to find occurrences of the old host names and changes them to the new names. When it is finished, the tool produces a report of the changes.

You must run the Update Host Name References tool on every SAS server host machine and every middle-tier host machine in the deployment, even if only one host name has changed. The reason for this requirement is that each machine's configuration files contain references to other machines in the deployment. You must also run the tool on client machines where certain software is installed. See "Client Software Components That Require Changes to Host Name References" on page 481. Otherwise, it is not necessary to run the tool on client machines.

In each execution of the tool, you must enter the complete list of host name changes for the deployment.

Note: You can automate running the SAS Deployment Manager when you need to perform the same configuration action on multiple machines in your deployment. The SAS Deployment Manager uses the same record and playback mechanism as the SAS Deployment Wizard to perform a non-interactive, silent configuration. For more information, see "Automating the SAS Installation on Multiple Machines" in *SAS Intelligence Platform: Installation and Configuration Guide*.

After running the Update Host Name References tool, you might still need to manually change some host name references. For example, if the name of the metadata server host machine has changed, you might need to make manual changes to connection profiles on client machines. These and other manual steps are specified in the generated report and in "Step 10: Complete the Required Manual Steps" on page 494.

Before using the Update Host Name References tool, be sure to review all of the documentation in this appendix and complete all of the specified prerequisite tasks.

When to Use the Update Host Name References Tool

Use Case 1: An Existing Host Machine Name or Network Domain Name Has Changed

The Update Host Name References tool can be used when the host name (or network domain name) of one or more existing SAS server or middle-tier host machines has changed. In this situation, you can use the tool to update your SAS deployment to refer to the new host name. The following examples illustrate this use case:

*   The name of the machine on which a single-machine deployment is installed has changed, as shown in the following example. You can use the Update Host Name References tool to change all occurrences of the machine name in the deployment to the new name.

Old host name: HOST01.example.com
New host name: HOST99.example.com

*   In a multiple-machine deployment, the name of a machine on which a SAS server is installed has changed, as shown in the following example. You can use the Update Host Name References tool to change all occurrences of the server machine's name in the deployment to the new machine name.

SAS Metadata Server host machine

SAS Application Server host machine

Middle tier host machine

MET01.example.com

Old name: APP01.example.com
New name: APP99.example.com

MID01.example.com

*   The name of the network domain in which a SAS deployment has been installed has changed, as shown in the following example. You can use the Update Host Name References tool to change all occurrences of fully qualified host names throughout the deployment to incorporate the new domain name.

SAS Metadata Server host machine

SAS Application Server host machine

Middle tier host machine

Old names: MET01.example.com   APP01.example.com   MID01.example.com
New names: MET01.orionstarsports.com   APP01.orionstarsports.com   MID01.orionstarsports.com

Use Case 2: One or More Servers Have Been Cloned to New Machines

The Update Host Name References tool can be used when you have created an exact clone of one or more SAS servers or middle-tier machines on another machine or machines with different host names. The following examples illustrate this use case:

• You have acquired a new machine and you want to move one of the servers in your deployment to the new machine.

• You want to create an exact clone of your entire deployment on another host machine, on another virtual machine within the same host, or on another set of host machines. For example, you might be creating a test deployment that mirrors an existing production deployment.

To create the clone, you must use cloning, ghosting, or virtualization software to transfer full-system images. On Windows, use a utility such as Norton Ghost from Symantec. On UNIX, use the cloning tool that is appropriate for your flavor of UNIX. After cloning the machine, you can use the Update Host Name References tool to update your SAS deployment with the new machine's host name.

The following are some specific examples of this use case:

• A single-machine deployment has been cloned to another machine, as shown in the following example. You can use the Update Host Name References tool in the new deployment to update all occurrences of the host name to the new host name.
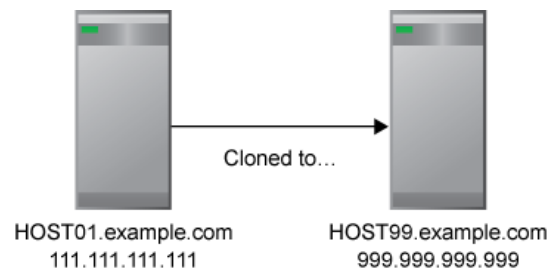
• In a multiple-machine deployment, cloning software has been used to move the metadata server to a new machine, as shown in the following example. You can use the Update Host Name References tool to change all occurrences of the metadata server's host name so that they refer to the new metadata server host machine.
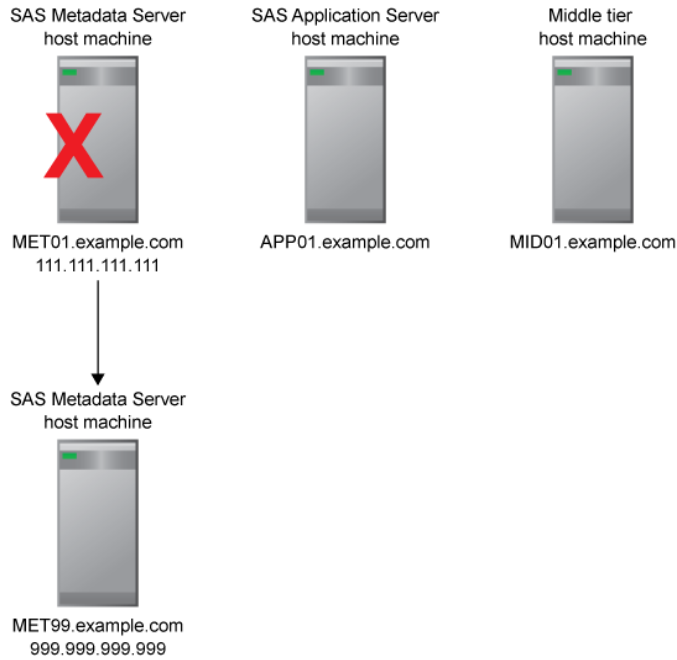
- Cloning software has been used to create a mirror of an entire multiple-machine deployment on a different set of machines, as shown in the following example. You can use the Update Host Name References tool in the new deployment to change all occurrences of each machine name to refer to the new machine names.
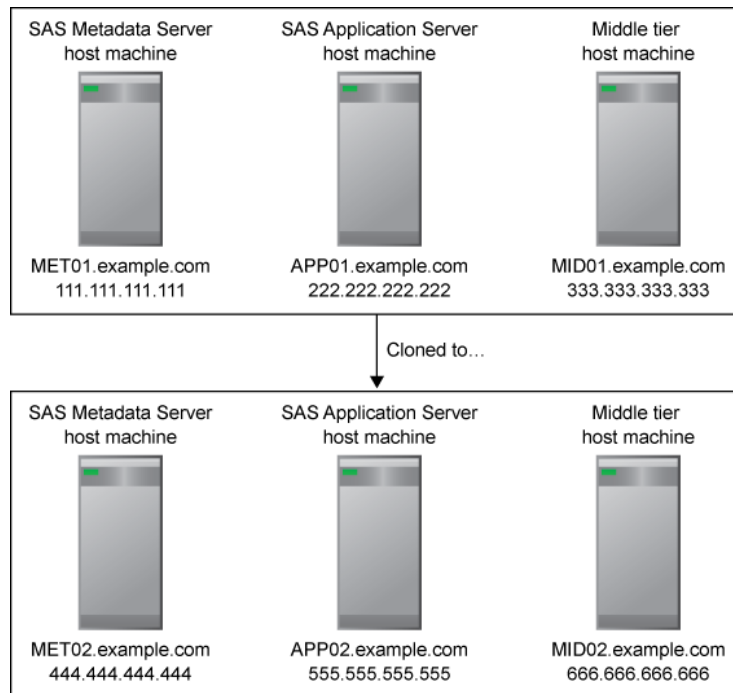
When Not to Use the Update Host Name References Tool

The Update Host Name References tool is intended only for the use cases that are described in "When to Use the Update Host Name References Tool" on page 477. The tool does not do the following:

- It does not copy software, configuration information, data, or metadata from one machine to another.

- It does not update anything in your deployment except host name references.

- It does not support moving a server to a machine that is of a different operating system or machine type than the existing server machine. It supports only scenarios in which ghosting, cloning, or virtualization software has been used to set up the new machine.

  CAUTION:
  > **If you use a method other than ghosting, cloning, or virtualization software to set up a new machine, then the use of the Update Host Name References tool is not supported for that machine.** For example, use of the Update Host Name References tool is not supported if you set up a new machine by running the SAS Deployment Wizard and then copying SAS configuration files from an existing machine.

The following table lists tools that are available to update various aspects of your deployment. Use this table to make sure that you have selected the correct tool for your situation.

Table 33.1   *Tools for Updating Your Deployment*

Tool Name	Purpose	Documentation Reference
Update Host Name References (in SAS Deployment Manager)	Find references to one or more server or middle-tier host names in a deployment, and replace them with new host names. Use this tool after the name of a SAS server or middle-tier host machine has changed, or after one or more server or middle-tier machines have been cloned to a new machine using ghosting, cloning, or virtualization software.	Chapter 33, "Using the SAS Deployment Manager to Update Host Name References," on page 475
SAS Deployment Wizard	Install and configure SAS software. Use this tool if you want to move a SAS Application Server to a different type of machine or a machine with a different operating system (in which case you cannot use ghosting, cloning, or virtualization software).	"Defining Multiple Application Servers" in *SAS Intelligence Platform: Application Server Administration Guide*
Promotion tools (Export SAS Package and Import SAS Package)	Promote individual objects or groups of objects from one location to another, either on the same metadata server or a different metadata server. You can also include associated physical content in the promotion.	Chapter 20, "Promotion Tools Overview," on page 265 and SAS Management Console Help
SAS Migration Utility	Populate a SAS deployment with metadata, associated physical data, and configuration information from an existing deployment.	"SAS Migration Utility Reference" in *SAS Intelligence Platform: Migration Guide*

Client Software Components That Require Changes to Host Name References

You must run the Update Host Name References tool on client machines if any of the following software components are installed:

- SAS Customer Intelligence Plug-ins for SAS Management Console

- SAS Customer Intelligence Studio

- SAS Customer Intelligence Utilities

- SAS Grid Manager Client Utility

- SAS Marketing Automation Integration Utilities

- SAS Marketing Automation Launcher

- SAS Marketing Optimization Client

If a client machine does not contain any of these components, then you do not need to run the Update Host Name References tool on the machine. However, if the name of the metadata server host machine has changed, you might need to make manual changes to connection profiles on client machines. See "Step 10: Complete the Required Manual Steps" on page 494.

What Modifications Are Performed by the Update Host Name References Tool?

For each existing machine whose name has been changed (or each machine that you are cloning to a new machine), the Update Host Name References tool enables you to specify the old host name and the new host name. The tool searches metadata, configuration files, and other files to find occurrences of the old host names that you specify. It changes these names to the new names that you specify and then produces a report of the changes. Modifications are performed in the following locations, as necessary:

- files in the SAS configuration directory, such as the following:

  - SAS server configuration (sasv9.cfg) files.

  - omaconfig.xml and metaparms.sas, which are used by the metadata server.

  - the SAS object spawner's start-up script and metadataConfig.xml file.

  - XML files that are used by SAS Deployment Tester.

  - properties files and configuration files for remote and local services, if your configuration includes the SAS middle tier.

  - login configuration files for the Java Authentication and Authorization Service (JAAS), if your configuration includes the SAS middle tier.

  - files that contain Windows user IDs that are qualified by a host name (for example, host1\sasadm or sasadm@host1). If you use this type of account for the unrestricted user, trusted user, or other users whose IDs are stored in configuration files, then the host names in the user IDs are updated as needed. These updates affect some of the previously listed files, as well as the following additional files:

    - adminUsers.txt and trustedUsers.txt, which are used by the metadata server.

- sasv9_meta.cfg, which is used by the SAS OLAP Server and the SAS/SHARE server.

- the SAS/CONNECT spawner's metadataConfig.xml file.

- JDBC data source definition files, if your configuration includes the middle tier.

- configuration files for the Stored Process web application, if your configuration includes the middle tier. The WAR and EAR files for the application are then rebuilt. The Stored Process web application is redeployed automatically.

- metadata properties such as the following:

  - properties in server metadata. For example, host names are stored in the server connection properties for SAS Pooled Workspace Servers, as shown in this example from the Server Manager tree in SAS Management Console:
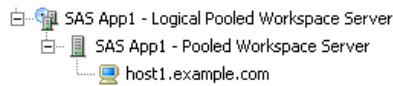
  - properties in metadata for SAS middle-tier services. For example, host names are stored in the metadata properties for SAS Web Infrastructure Platform Services, as shown in this example from the Configuration Manager plug-in for SAS Management Console:
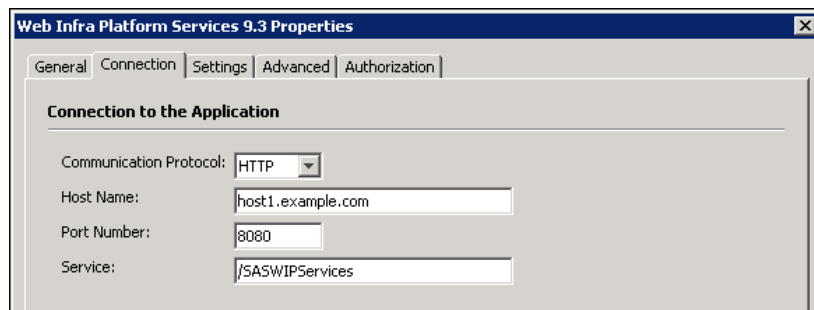
  - the SAS object spawner service. If this service is present on a Windows machine, the tool deletes the service and re-creates it using the new host name.

    Note: The SAS object spawner service is the only service that includes a host name reference. The other SAS services do not need to be re-created.

Additional files and metadata might also be changed, depending on the software that you have installed. After you run the tool, a report is generated that lists all of the metadata properties and files that were changed.

Modifications Not Performed by the Update Host Name References Tool

The Update Host Name References tool does not modify host names in the following locations in your deployment:

- certificates for Transport Layer Security (TLS), if TLS has been configured for SAS Web Server. Instructions for changing host name references in these certificates are provided in the report that is generated when you run the tool.

- log files. It is not necessary to update host names in log files, since these files contain historical information that is not used in processing.

- binary files, temporary files, and SAS data sets. Any files of these types that are found in the SAS configuration directory are ignored.

- user IDs for Windows accounts that are stored in metadata and are qualified with a machine name. Accounts that are associated with the SAS General Servers group (typically *host-name*\sassrv) are examples. These user IDs must be updated manually using SAS Management Console.

- users' connection profiles on client machines (unless the client software is listed in "Client Software Components That Require Changes to Host Name References" on page 481). For other client software, connection profiles must be updated manually on each client machine if the metadata server's host name has changed.

- the sas-environment.xml file in **SAS-configuration-directory/Lev1/Web/ Common** on middle-tier machines. This file is not updated automatically because it might contain site-specific customizations that need to be preserved.

- multicast parameters on middle-tier machines. If you have created a new deployment that will be running concurrently with your original deployment, then you might need to modify these parameters to avoid potential conflicts.

- Windows user IDs that are associated with scheduled reports. If the user ID is qualified by a machine name or network domain name that has changed, then you will need to reschedule the report after running the Update Host Name References tool.

For information about making manual changes to client profiles, the sas-environment.xml file, and multicast parameters, see "Step 10: Complete the Required Manual Steps" on page 494 .

The report that is generated by the Update Host Name References tool might list other files that were not updated by the tool.

Process for Updating Host Name References

Summary of the Update Host Name References Process

Follow these steps to use the Update Host Name References tool:

1. Document your host name changes.

2. Back up your current deployment.

3. Check the unrestricted user account (Windows only).

4. Change existing names at the operating system level, or set up new machines to be identical to the current machines.

5. Update the metadata server's backup location, if necessary..

6. Back up the newly cloned metadata server.

7. Identify the machines on which to run the tool, and determine the execution order.

8. Run the Update Host Name References tool on each identified machine.

9. Rebuild the Windows service definition for each SAS Web Application Server instance on page 491.

10. Complete required manual steps.

11. Update host name references in Platform Suite for SAS, if applicable.

12. Start your servers and perform validation tasks.

13. Make a complete backup of the updated deployment.

Note: If you plan to run any of the other SAS Deployment Manager functions on a new machine in addition to running the Update Host Name References tool, be sure to run the Update Host Name References tool first. For example, if you want to run both the Update Host Name References tool and the Update Passwords tool, you must run the Update Host Name References tool first.

Note: You can use command line options to run SAS Deployment Manager in quiet mode or record and playback mode. For details, see *SAS Deployment Wizard and SAS Deployment Manager 9.4: User's Guide*, available at http://support.sas.com/deploywizug94.html.

Step 1: Document Your Host Name Changes

For each machine whose machine name is to change (or for each machine that you are cloning), write down the current name and the new name. List all forms of each name, including the following:

- the fully qualified host name (for example, P4001.example.com)

- the short host name (for example, P4001)

- the IP name (for example, 111.111.111.111)

Step 2: Back Up Your Current Deployment

Make a complete backup of your current deployment. The backup should include the metadata repositories, as well as the file systems on each machine. See Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153 and Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

Step 3: Check the Unrestricted User Account (Windows Only)

When you run the Update Host Name References tool, you will be prompted for a user ID and password for an unrestricted user that has either an internal account (for example, sasadm@saspw) or a network account. If the unrestricted user has only an external account that is qualified by a local machine name that is changing, you must set up a new account before running the tool. See "Creating a New Account for the Unrestricted User (Windows Only)" on page 504.

Step 4: Change Existing Names at the Operating System Level, or Set Up New Machines to Be Identical to the Current Machines

If you are renaming one or more existing server or middle-tier machines (or renaming a network domain), then you must change the names at the operating system level before you run the Update Host Name References tool.

If you are creating a clone of one or more SAS servers or middle-tier machines (or of an entire SAS deployment), you must use cloning, ghosting, or virtualization software to transfer full-system images. On Windows, use a utility such as Norton Ghost from

Symantec. On UNIX, use the cloning tool that is appropriate for your flavor of UNIX. After the clone is created, verify that the cloned machine is identical to the source machine (except for the machine name). For example, the SAS configuration directory and its contents, the SAS installation directory and its contents, user accounts, and third-party software must be identical.

CAUTION:

> **If you use a method other than ghosting, cloning, or virtualization software to set up a new machine, then the use of the Update Host Name References tool is not supported for that machine.** For example, use of the Update Host Name References tool is not supported if you set up a new machine by running the SAS Deployment Wizard and then copying SAS configuration files from an existing machine.

After you have finished the cloning process, do not change anything on the machine before you run the Update Host Name References tool. For example, do not change any directory names or user accounts.

Step 5: Update the Metadata Server's Backup Location, If Necessary

You might need to specify a new location for metadata server backups in the following situations:

- The current backup location is on a machine that has been renamed or cloned.

- You have cloned the metadata server machine (or machines), or you have cloned the entire deployment. Specifying a new backup location for the newly cloned deployment is especially important if the old deployment will be running concurrently with the new deployment.

Follow these steps:

1. Log on to SAS Management Console. If the metadata server has been cloned to a new machine, connect to the new machine.

2. Select **Metadata Manager ⇨ Metadata Utilities ⇨ Server Backup ⇨ Backup Configuration**.

3. In the Backup Configuration, update the pathname.

   `TIP` If you have cloned the metadata server machine (or machines), or you have cloned the entire deployment, do not specify the same backup location that was used in the original deployment.

   `TIP` For backup location requirements, see "Modifying the Metadata Server Backup Configuration (Backup Location and Retention Policy)" on page 175.

Step 6: Back Up the Newly Cloned Metadata Server

If you have cloned the metadata server machine (or machines), make a backup of the newly cloned server. See Chapter 12, "Backing Up and Recovering the SAS Metadata Server," on page 169.

Step 7: Identify the Machines on Which to Run the Tool and Determine the Execution Order

Next, identify the machines on which you need to run the Update Host Name References tool, and then determine the order in which you need to execute the tools on those machines. The requirements are as follows:

- The tool must be run on every SAS server host machine and every middle-tier host machine in the deployment, even if only one host name has changed. The reason for this requirement is that each machine's configuration files contain references to other machines in the deployment.

- The tool must also be run on client machines where certain client software is installed. See "Client Software Components That Require Changes to Host Name References" on page 481. Otherwise, it is not necessary to run the tool on client machines.

- Run the tool on machines in the following order:

  1. The SAS Metadata Server machine. If you are using metadata server clustering, run the tool first on the initially configured metadata server node, and then on each additional node.

  2. Each SAS Application Server machine.

  3. Each middle-tier machine. If you have a horizontal middle-tier cluster, run the tool first on the primary horizontal middle-tier cluster machine, and then on each middle-tier cluster node.

  4. Any required client machines.

For example:

- If the name of one or more existing machines has changed (or the name of the network domain in which the deployment is installed has changed), then run the tool on each machine in the deployment. Run the tool on the metadata server host machine (or machines) first, then on the SAS application servers host machines, and then on the middle-tier host machine (or machines).

- If you have cloned an entire deployment to one or more new machines, then run the tool on each machine in the new deployment. In a multiple-machine deployment, run the tool on the new metadata server host machine (or machines) first, then on the new SAS Application Server host machines, and then on the new middle-tier host machine.

- If you have cloned just part of a multiple-machine deployment to new machines, then run the tool on each of the new machines as well as on any machines that were not cloned to new machines.

  For example, if you have cloned the metadata server host to a new machine, then begin by running the tool on the new metadata server host machine (or machines). Then run the tool on the existing application server host machines, and then on the existing middle-tier host machine.

- If you have identified client machines on which you need to run the tool, run it on those machines last, after the server and middle-tier machines.

For further guidance in identifying the machines and determining the required order, see "Example Scenarios for Updating Host Name References" on page 498.

Step 8: Run the Update Host Name References Tool on Each Identified Machine

Before running the Update Host Name References tool, make sure you have completed all of the preparatory steps, as appropriate for your deployment. These include Step 1: Document Your Host Name Changes through Step 7: Identify the Machines on Which to Run the Tool and Determine the Execution Order.

CAUTION:

> **You must run the tool on every SAS server host machine and middle-tier machine in the deployment, even if only one machine name has changed. Run the tool on the metadata server first, then on SAS Application Server machines, then on the middle-tier machine, and then on any required client machines.** If you are using metadata server clustering, run the tool separately on each node in the cluster, beginning with the initially configured metadata server. If you have a horizontal middle-tier cluster, run the tool first on the primary horizontal middle-tier cluster machine, and then on each middle-tier cluster node.

On each machine, follow these steps to run the Update Host Name References tool:

1. Make sure that the servers are in the appropriate state, as described here:

Server	Required State
SAS Web Infrastructure database (usually the SAS Web Infrastructure Platform Data Server)	Must be started.
SAS Metadata Server (non-clustered)	Must be stopped if you are running the Update Host Name References tool on the metadata server host machine. Must be started if you are running the tool on any other machine.
Clustered SAS Metadata Server	All nodes must be stopped if you are running the Update Host Name References tool on any one of the cluster nodes. All nodes must be started if you are running the tool on any other machine.
All other server processes on all machines	Must be stopped. These include SAS server processes as well as middle-tier processes such as the SAS Web Server, SAS Web Application Server, and the SAS Environment Manager server.

CAUTION:

> **If you have cloned any of these servers (or the entire deployment) to new machines, then as a best practice you should either stop or pause the old servers. The old metadata server must be stopped.** You will be able to start the servers on the new machines even though the host name references have not been updated in the deployment.

CAUTION:

> **If you have cloned the middle-tier machine (or the entire deployment) to a new machine, then you must also stop the middle-tier processes on the old middle-tier machine (or in the old deployment).** If the two middle tiers will be running concurrently, then you might need to modify the multicast parameters before restarting these processes, as described in "Step 10: Complete the Required Manual Steps" on page 494.

2. Log on to the machine with the same user account that was used to deploy SAS 9.4. (On Windows, the user account must have Administrator rights.) Start the SAS Deployment Manager on the machine by navigating to *SAS-installation-directory*/**SASDeploymentManager/9.4** and launching **sasdm.exe** (Windows), **sasdm.sh** (UNIX) or **sasdm.rexx** (z/OS).

3. On the first page of the SAS Deployment Manager, select **Update Host Name References**.

4. On the Update Host Name References Information page, review the information that is displayed, which is a summary of the information in this documentation.

5. On the Select Configuration Directory/Level page, select the configuration directory for which you are running the tool.

   If the configuration directory is not listed, select **Enter Configuration Directory and Level** and use the **Browse** button to find the directory. Then select the appropriate configuration level from the drop-down box.

6. On the Specify Connection Information page, enter the following information:

   • the metadata server host machine and port number.

   If the name of the metadata server machine has changed, or if you have cloned the metadata server (or the entire deployment) to a new machine, enter the new metadata server machine name at the **Host Name** prompt.

   If you are running the tool on a metadata server cluster node, enter the new machine name for the first (initially configured) node.

   • the user ID and password of an unrestricted user (that is, a user who is in the **SAS Metadata Server: Unrestricted** role).

   Note: If the metadata server has been cloned or renamed, and if the unrestricted user has only a local machine account, see "Step 3: Check the Unrestricted User Account (Windows Only)" on page 484.

7. On Specify Host Information page, make the following entries:

   Old Host Name(s)
   Enter a comma-delimited list of the host names for the machines whose names have changed, or for the machines that have been cloned to new machines. Be sure to include all machines in the deployment whose names have changed (or that have been cloned to new machines). Because the tool is not case-sensitive, you can enter the names using either uppercase or lowercase letters.

   Fully qualified host names, short names, and IP addresses can be specified. To determine which name formats to include, see Table 33.2 on page 489.

   New Host Name(s)
   Enter a comma-delimited list of the new host names for the machines whose names have changed (or the machines on which you have created clones of existing machines). Be sure to include the same forms of the machine names and to enter the names in the same order that you entered the corresponding names in the **Old Host Name(s)** field.

Note: You cannot specify a given machine name in both **New Host Name(s)** and **Old Host Name(s)**. For example, `host01.mycompany.com` cannot be entered in both fields.

Perform DNS lookup

For a given host machine, different files and metadata in your deployment might contain different forms of the host name, including the fully qualified name, the short name, and the IP address. If you are cloning one or more servers to a new machine, then you can select **Perform DNS lookup** option, which accesses the Domain Name System to determine all of the forms of each host name.

If the name of an existing host machine (or network domain name) has changed, then you must deselect **Perform DNS lookup**. The Domain Name System does not contain information about machine names or network domains that no longer exist.

CAUTION:

You must complete the preceding fields in such a way that all applicable forms of the host name are updated correctly throughout your deployment.

To make sure that all applicable forms of the machine names are updated correctly, refer to the following table:

Table 33.2 *Determining the Correct Format for Host Name Entries*

Scenario	Format for Entering Old and New Host Names	Perform DNS Lookup?
One or more existing SAS server or middle-tier host machines have changed.	Enter both the fully qualified name and the short name for each machine whose name has changed. It is not necessary to enter IP addresses, since the IP address stays the same in this scenario. *Note:* If your metadata includes Windows user accounts that are qualified with host names (for example, *host-name*\sassrv), be sure to include short host names in the list.	Deselect **Perform DNS lookup**.
The name of a network domain name has changed.	Enter fully qualified host names for the machines that contain the network domain name that is changing. It is not necessary to enter short names or IP addresses, since these forms of the name stay the same in this scenario.	Deselect **Perform DNS lookup**.
One or more servers (or an entire deployment) have been cloned to a new machine or machines.	Enter fully qualified host names for the machines that have been cloned to new machines. If you are selecting **Perform DNS lookup** (which is strongly recommended), then you do not need to enter the machines' IP addresses or short names.	As a best practice, select **Perform DNS lookup**. Otherwise, you must enter all forms of the machine names.

For example entries, see "Example Scenarios for Updating Host Name References" on page 498.

If you select **Perform DNS lookup**, then the lookup is performed when you click **Next**. The results of the lookup appear on the next page (Update Host Name References Summary).

If the specified names are not found in the Domain Name System, the following message appears: **Unable to resolve the DNS host name for** *machine-name*. **Enter a different value for the host name or clear the option to perform a DNS lookup of the name.** You can then correct your entry and click **Next** to repeat the lookup. If you are changing the name of an

existing machine or network domain, then you must deselect **Perform DNS lookup**. The lookup cannot be performed in these scenarios.

8. On the Update Host Name References Summary Information page, review the list of host name changes that will be made, and make sure that they match the old and new names that you identified in "Step 1: Document Your Host Name Changes" on page 484. The current host names are listed in the first column, and the new host names are listed in the second column.

   If you did not select the **Perform DNS lookup** option, this page lists all host name changes exactly as you specified them on the previous page.

   If you selected the **Perform DNS lookup** option, this page lists all forms of the host name that were found during the DNS lookup, along with the values that will replace those names. For example, if you specified P4001.orionstarsports.com, P4002.orionstarsports.com, and P4003.orionstarsports.com as the old host names and T4001.orionstarsports.com, T4002.orionstarsports.com, and T4003.orionstarsports.com as the new host names, this page might display a list similar to the following:
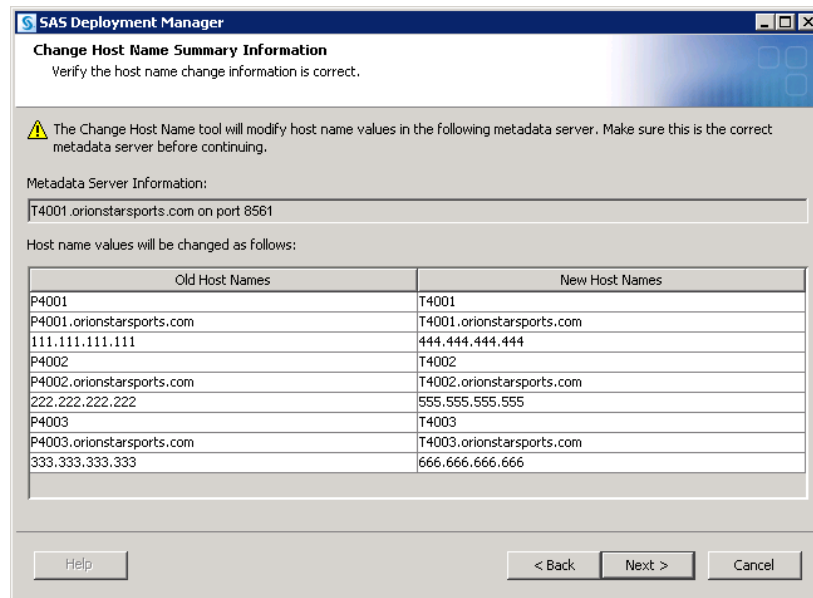
   In the preceding example, the DNS lookup option found all other forms of the machine names for the old machine, including the short names and the IP addresses. These names are listed in the left column. The DNS lookup option also found the corresponding names for the new machines and listed them in the right column.

   Review the information carefully to ensure the following:

   • The correct metadata server is specified in the Metadata Server Information.

   • The correct host name changes are listed.

   If the information is correct, click **Next** to proceed. Otherwise, click **Back** to make the necessary corrections.

9. The Summary page lists the main execution steps, including a list of the products on the machine whose configuration information will be changed. If you are sure you are ready for the host names to be changed, click **Start** on this page.

   CAUTION:
   The Update Host Name References tool does not have an "undo" feature. The results of running the tool can be undone only by restoring the machine

> **from a backup.** Therefore, you should click **Start** only if you are sure you have set up the execution correctly.

10. The next page displays the status of each step as it is executed.

11. When execution is finished, the tool creates a document called ChangeHostName_*date-and-time*.html. This document lists all of the files and metadata properties that were changed and provides instructions for any additional steps to be performed manually. It might also report warnings if issues were encountered during the execution.

    If a browser is installed on your machine, the report is displayed automatically. Otherwise, you can retrieve the report from the **Documents** directory within your configuration directory.

12. If you are running the tool on the SAS Metadata Server machine, the tool restarts the metadata server and initiates a metadata server backup.

    If you are running the tool on a node of a SAS clustered metadata server, the tool starts the node with the -recover option. This causes the node to be recovered from the backup of the first node and picks up any changes the tool might have made.

    Note: The -recover option generates an alert email, which can be ignored in this situation.

Repeat this process on each machine that you identified in , being sure to specify exactly the same old and new host names that you specified in the first execution. If you have multiple SAS configuration directories (for multiple SAS deployments) on the same machine, you must run the Update Host Name References tool separately for each directory.

If any part of the execution fails, see .

Step 9: Rebuild the Windows Service Definition for Each SAS Web Application Server Instance

Note: This step is applicable to the February 15, 2022 and later releases of SAS 9.4M7 and SAS 9.4 M8 on Windows environments.

Prior to the February 15, 2022 release of SAS 9.4M7, SAS Web Application Server instances are managed with Pivotal tc Server. Thereafter, SAS Web Application Server instances are managed with Apache Tomcat. In releases where SAS Web Application Server instances are managed with Apache Tomcat, you must rebuild the Windows service definition for each instance of the SAS Web Application Server after updating the host name references or updating the wrapper.conf files. This is for Windows environments only.

SAS Web Application Server instances on Windows are managed with the SASServiceCtl.ps1 Windows PowerShell script. Rebuild the Windows service definition for each instance of the SAS Web Application Server with this script as follows:

1. Open a Windows command prompt from which you will run the commands in the following steps.

2. Navigate to the following directory that contains the SASServiceCtl.ps1 script:
   \SAS-configuration-directory\Levn\Web\WebAppServer \SASServer*N*_*M*\bin

   Note: There is a copy of the SASServiceCtl.ps1 script in this directory for each SAS Web Application Server instance.

3. List each SAS Web Application Server instance.

   a. From Windows Services Manager, find the SAS Web Application Server instances, such as SASServer1_1 and SASServer2_1.

   b. Determine a common string that appears in the **Name** column for each SAS Web Application Server instance, for example, **WebAppServer**.

      Note: This string is used in the command to list all of the SAS Web Application Servers that contain that string in the service name.

   c. Run the following command to list the server names of each SAS Web Application Server instance, substituting the common string that you identified in the last step as the value of the NAME option:

      ```
      powershell -file .\SASServiceCtl.ps1 -action list -name
      WebAppServer
      ```

   d. Make a note of the **Service Name** and **Service Display Name** values for each SAS Web Application Server instance, as they are needed for subsequent commands.

4. Run the setenv.bat script to set the CATALINA_HOME environment variable, as follows: **setenv.bat**

   Note: The UNINSTALL option requires the existence of the CATALINA_HOME environment variable that defines where the core of the Tomcat software is installed.

5. Remove each instance of the SAS Web Application Server. For the value of the NAME parameter, use the name of the SAS Web Application Server. This name appears in the **Service Name** column from the resulting table in step 3 above.

   ```
   powershell -file .\SASServiceCtl.ps1 -action uninstall
   -name WebAppServer
   ```

   For example, here is the command to uninstall the SAS Web Application Server instance named SAS-Config-Lev1-SASServer1_1-WebAppServer.

   ```
   powershell -file .\SASServiceCtl.ps1 -action uninstall
   -name SAS-Config-Lev1-SASServer1_1-WebAppServer
   ```

   Note: Run the command above for each instance of the SAS Web Application Server, substituting the correct name of the other SAS Web Application Server instances for the value of the NAME option..

6. Install the first SAS Web Application Server instance.

   a. Run the setenv.bat script to set the CATALINA_BASE environment variable for your first SAS Web Application Server instance.

      Note: The INSTALL option requires the existence of the following environment variables:

CATALINA_HOME	Defines where the core of the Apache Tomcat software is installed.
CATALINA_BASE	Defines the location where the server instance is to be installed.

      For example, suppose that the name of your first SAS Web Application Server instance is SAS-Config-Lev1-SASServer1_1-WebAppServer, and you are

currently in the following directory for this instance: **\SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer1_1\bin**.

Run the setenv.bat script from this directory as follows: **setenv.bat**

b. Install the first SAS Web Application Server instance. For the value of the NAME parameter, use the name of the SAS Web Application Server. This name appears in the **Service Name** column from the resulting table in step 3 above.

```
powershell -file SASServiceCtl.ps1 -action install -name
WebAppServer
```

For example, here is the command to install the first instance of the SAS Web Application Server named SAS-Config-Lev1-SASServer1_1-WebAppServer. This command also includes the DISPLAY and DESCRIPTION options (with sample values) on the INSTALL command to specify the correct display name and description for that Windows service.

```
powershell -file .\SASServiceCtl.ps1 -action install -display
 "SAS [Config-Lev1] SASServer1_1 - WebAppServer" -description
"SASServer1_1 WebAppServer at Config-Lev1 on port 8080"-name
 SAS-Config-Lev1-SASServer1_1-WebAppServer
```

7. Install the other SAS Web Application Server instances.

a. Run the setenv.bat script to set the CATALINA_BASE environment variable for your other SAS Web Application Server instances. The CATALINA_BASE environment variable must be set for each SAS Web Application Server instance. Therefore, you must run the script from the following directory for each SAS Web Application Server instance: **SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer*N_M*\bin**

For example, suppose that the name of your second SAS Web Application Server instance is SAS-Config-Lev1-SASServer2_1-WebAppServer, and that the directory for this instance is **SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer2_1\bin**. Also, suppose that you are currently in the following directory for the first SAS Web Application Server instance: **\SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer1_1\bin**.

Run the setenv.bat script for your second SAS Web Application Server instance directly from the directory of the second SAS Web Application Server instance as follows:

```
..\..\SASServer2_1\bin\setenv.bat
```

Note: In step c below, you are directed to repeat this step for the other SAS Web Application Server instances. Substitute the directory name of the other SAS Web Application Server instances for SASServer2_1.

b. Install the other SAS Web Application Server instances:

```
powershell -file SASServiceCtl.ps1 -action install -name
WebAppServer
```

For example, here is the command to install the instance of the SAS Web Application Server named SAS-Config-Lev1-SASServer2_1-WebAppServer. This command also includes the DISPLAY and DESCRIPTION options (with sample values) on the INSTALL command to specify the correct display name and description for that Windows service.

```
powershell -file SASServiceCtl.ps1 -action install -display
"SAS [Config-Lev1] SASServer2_1 - WebAppServer" -description
```

```
"SASServer2_1 WebAppServer at Config-Lev1 on port 8180" -name
 SAS-Config-Lev1-SASServer2_1-WebAppServer
```

c. Repeat steps a and b for the remaining SAS Web Application Server instances. Make sure to substitute the correct name of the other SAS Web Application Server instances.

Step 10: Complete the Required Manual Steps

In addition to listing the files and metadata profiles that were updated, the ChangeHostName_*date-and-time*.html report provides instructions for steps to be performed manually. The report is stored in the **Documents** directory within your configuration directory. The manual steps might include the following:

*   updating SAS Environment Manager agents and resource tables

*   updating TLS certificates, if you are using TLS for the SAS Web Server

*   redeploying the Stored Process web application, if the application's configuration files contain Windows user IDs that are qualified by a host name

*   restarting SAS servers

You might need to perform the following additional steps:

*   Update user IDs for Windows accounts that are stored in metadata and are qualified with a host name. Accounts that are associated with the SAS General Servers group (typically *host-name*\sassrv) are examples. These accounts are used to launch stored process servers and pooled workspace servers. Open User Manager in SAS Management Console, right-click the entry for the **SAS General Servers** group, and select **Properties**. Select the **Accounts** tab, as shown here:
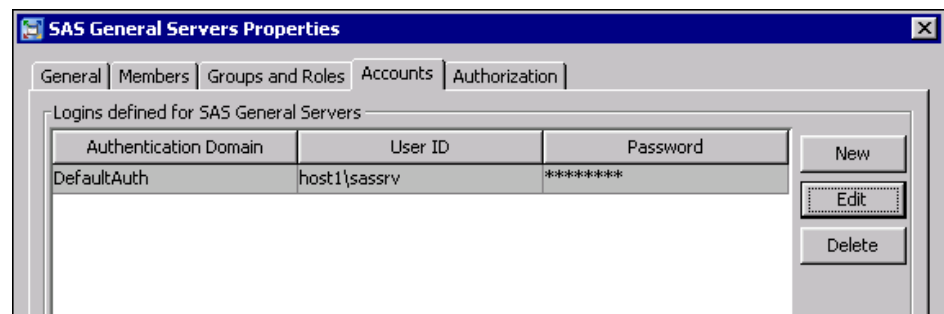
In the user ID, replace the old host name with the new host name.

*   If the metadata server's host name has changed, then you must update host names in users' connection profiles on each machine where client software is installed (unless the client software is listed in "Client Software Components That Require Changes to Host Name References" on page 481). Be sure to update the profiles for SAS Management Console, which is typically installed on server and middle-tier machines.

    For instructions, see "Administering Connection Profiles for Desktop Applications" in *SAS Intelligence Platform: Desktop Application Administration Guide*.

*   If you have created a new deployment that will be running concurrently with your original deployment, and if the deployments include a middle tier, then you might need to modify the multicast parameters to avoid potential conflicts. See "Administer Custom Applications" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

- If your deployment includes a middle tier, then you might need to update host names in the sas-environment.xml file. This file is located on middle-tier machines in the following path: **SAS-configuration-directory/Lev1/Web/Common**. The file is not updated automatically because it might contain site-specific customizations that need to be preserved. For more information, see "Configure the SAS Environment File" in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

- If you have previously scheduled a report that is associated with a machine or domain-qualified user ID on Windows, and the machine name or network domain name has changed, then you will need to reschedule the report.

- If you use Platform Suite for SAS for scheduling, update the host name references in this software as described in "Step 11: Update Host Names in Platform Suite for SAS".

Step 11: Update Host Names in Platform Suite for SAS

About Updating Host Names in Platform Suite for SAS

If you use Platform Suite for SAS for scheduling, the Update Host Name References tool automatically updates host names that exist in metadata for scheduling servers and grid resources. However, you must manually update the host name references in Platform Suite for SAS. The following topics provide instructions:

- "Update Host Names in Platform Suite for SAS: Windows Instructions"
- "Update Host Names in Platform Suite for SAS: UNIX Instructions"
- "Update Host Name References for Grid Resources"

Update Host Names in Platform Suite for SAS: Windows Instructions

On Windows systems, follow these steps to update host name references in Platform Suite for SAS:

1. Make sure that the Process Manager service and the LSF Cluster services have been stopped.

2. Update the host name references in the following files:

   - **JS_TOP\conf\js.conf**

   - **JS_TOP\conf\lsf.conf**

   - **JS_TOP\conf\lsf.cluster.cluster_name**

   - **JS_TOP\conf\ego\cluster_name\kernel\ego.conf**

3. Update the host name references in the following environment variables:

   - LSF_ENVDIR

   - EGO_CONFDIR

4. In the Windows registry, update the host name references in the values of the following keys:

   - **HKEY_LOCAL_MACHINE\SOFTWARE\Platform Computing Corporation\EGO\EGO_CONFDIR**

   - **HKEY_LOCAL_MACHINE\SOFTWARE\Platform Computing Corporation\EGO\EGO_SHAREDTOP**

- **HKEY_LOCAL_MACHINE\SOFTWARE\Platform Computing Corporation\LSF\LSF_ENVDIR**

5. Start the Process Manager service and the LSF services.

Update Host Names in Platform Suite for SAS: UNIX Instructions

On UNIX systems, follow these steps to update host name references in Platform Suite for SAS:

1. Make sure that the Process Manager service and the LSF Cluster services have been stopped.

   To stop the services, log on to the Process Manager server host either as Root or as the primary Process Manager administrator, and use the following commands:

   ```
   jadmin stop
   badmin hshutdown all
   lsadmin resshutdown all
   lsadmin limshutdown all
   ```

2. Update the host name references in the following files:

   - *JS_TOP***/conf/js.conf**

   - *JS_TOP***/conf/lsf.conf**

   - *JS_TOP***/conf/lsf.cluster.***cluster_name*

   - *JS_TOP***/conf/ego/***cluster_name***/kernel/ego.conf**

3. Update the host name references in the process manager and LSF environments, as follows:

   - If you are using the C shell (csh or tcsh), use the following scripts:

     - **source** *JS_TOP***/conf/cshrc.js**

     - **source** *JS_TOP***/conf/cshrc.lsf**

   - If you are using the Bourne shell (sh), the bash shell, or the Korn shell (ksh), use the following scripts:

     - **.***JS_TOP***/conf/profile.js**

     - **.***JS_TOP***/conf/profile.lsf**

4. Use the following commands to reconfigure the LSF cluster and restart the services:

   ```
   jadmin start
   lsadmin reconfig
   badmin reconfig
   badmin mbdrestart
   ```

Update Host Name References for Grid Resources

To update host names in grid resource specifications, follow these steps:

1. Log on to the grid control machine as the LSF administrator.

2. Issue the following command:

   ```
   addresource -r resource_name -m machine-name
   ```

   In *machine-name*, specify the new host name. If the name contains spaces, you must change the spaces to underscores.

3. Run the LSF commands to reconfigure the grid to recognize the new resource.

See Also
"Defining and Specifying Resources" in *Grid Computing in SAS*

Step 12: Start Servers and Perform Validation Tasks

After you have completed the necessary manual updates, start all of your servers as specified in ChangeHostName_*date-and-time*.html.

CAUTION:
If the unrestricted user or the SAS Trusted User has an external account that is qualified by a host name that has changed (for example, host1\sasadm or host1\sastrust), then you must stop and start the metadata server before you start any of the other servers. Doing so will ensure that the metadata server runs with the correct values for those accounts.

After the servers have been started, validate your deployment. For details, see:

- Chapter 6, "Operating Your Servers," on page 61

- "Validate the SAS 9.4 Servers" in *SAS Intelligence Platform: Installation and Configuration Guide*

You can also refer to your Instructions.html document, which is located in the **Documents** directory within your configuration directory. If you use this document, you will need to substitute new machine names where applicable.

Step 13: Make a Complete Backup of the Updated Deployment

After you have validated your servers, it is strongly recommended that you make a complete backup of your updated deployment. The backup should include the metadata repositories, as well as the file systems on each machine. See Chapter 11, "About Backing Up and Restoring Your SAS Content," on page 153 and Chapter 28, "Using the Deployment Backup and Recovery Tool," on page 387.

Troubleshooting the Update Host Name References Tool

The Update Host Name References tool creates a log for each software component whose metadata or configuration information needs to be updated. The logs are named *software-component*_changeHost_*date-and-time*.log and are written to **SAS-configuration-directory/Levn/Logs/Configure**. Refer to these logs if the update process for a particular software component completes with errors.

The SAS Deployment Manager also creates a log called SDM.log, which can be used for general troubleshooting for the SAS Deployment Manager. The log is written to the following path:

- On Windows Vista, Windows 7, and Windows Server 2008 systems: **C:\Users*user*\AppData\Local\SAS\SASDeploymentWizard\9.4**

- On other Windows systems: **C:\Documents and Settings*user*\Local Settings\Application Data\SAS\SASDeploymentWizard\9.4**

- On UNIX systems: **user's-home-directory/.SASAppData/SASDeploymentWizard/9.4**

Example Scenarios for Updating Host Name References

Example 1: The Machine Name Has Changed in a Single-Machine Deployment

In this example, a single-machine deployment is installed on HOST01.example.com, and the name of the machine has been changed to HOST99.example.com. All occurrences of the machine's fully qualified name and short name need to be updated throughout the deployment to refer to the new name.

Old host name: HOST01.example.com
New host name: HOST99.example.com

In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Change the host name of the machine at the operating system level.

3. Stop all of the servers, including the metadata server (or metadata server cluster).

4. Run the Update Host Name References tool on the machine.

   In the Specify Host Names dialog box, you must enter both the fully qualified host names and the short host names. (You cannot perform a DNS lookup when the name of an existing machine has changed.) You do not need to enter the IP address, because it remains the same in this scenario. Specify the following:

   • **Old host name: `HOST01.example.com,HOST01`**

   • **New host name: `HOST99.example.com,HOST99`**

   • **Perform DNS lookup**: Not selected

5. Perform the necessary manual steps, as listed in the report. Because the metadata server host name has changed, update the connection profiles for client applications where necessary.

6. Start the servers and perform the validation steps.

7. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Example 2: The Name of a SAS Application Server Host Machine Has Changed

In this example, the SAS Metadata Server, SAS Application Server, and SAS middle tier are installed on separate machines. The SAS Application Server is installed on a machine called APP01.example.com, and the name of this machine has been changed to APP99.example.com. All occurrences of the application server machine's fully qualified name and short name need to be updated throughout the deployment to refer to the new name.

SAS Metadata Server host machine	SAS Application Server host machine	Middle tier host machine
MET01.example.com	Old name: APP01.example.com New name: APP99.example.com	MID01.example.com

In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Change the host name of the SAS Application Server machine at the operating system level.

3. Stop all of the servers, including the metadata server (or metadata server cluster).

4. Run the Update Host Name References tool on MET01.example.com.

   In the Specify Host Names dialog box, you must enter both the fully qualified host names and the short host names. (You cannot perform a DNS lookup when the name of an existing machine has changed.) You do not need to enter the IP address, because it remains the same in this scenario. Specify the following:

   • **Old host name: `APP01.example.com,APP01`**

   • **New host name: `APP99.example.com,APP99`**

   • **Perform DNS lookup**: Not selected

5. Run the Update Host Name References tool on APP99.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 3.

6. Run the Update Host Name References tool on MID01.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 3.

7. Perform the necessary manual steps, as listed in the report.

8. Start the servers and perform the validation steps.

9. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Example 3: The Network Domain Name Has Changed

In this example, a SAS deployment is installed in a network domain called example.com, and the domain name has been changed to orionstarsports.com. All occurrences of fully qualified host names in the deployment need to be updated to incorporate the new domain name.
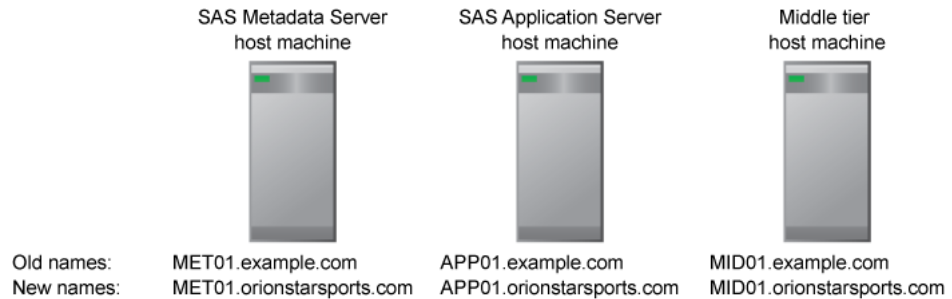
	SAS Metadata Server host machine	SAS Application Server host machine	Middle tier host machine
Old names:	MET01.example.com	APP01.example.com	MID01.example.com
New names:	MET01.orionstarsports.com	APP01.orionstarsports.com	MID01.orionstarsports.com

In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Stop all of the servers, including the metadata server (or metadata server cluster).

3. Run the Update Host Name References tool on MET01.orionstarsports.com.

   In the Specify Host Names dialog box, you need to enter only the fully qualified host names, because the short host names and the IP addresses remain the same in this scenario. You cannot perform a DNS lookup when the names of existing machines have been changed. Specify the following:

   • **Old host name:**
     MET01.example.com,APP01.example.com,MID01.example.com

   • **New host name:**
     MET01.orionstarsports.com,APP01.orionstarsports.com,MID01
     .orionstarsports.com

   • **Perform DNS lookup**: Not selected

4. Run the Update Host Name References tool on APP01.orionstarsports.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 3.

5. Run the Update Host Name References tool on MID01.orionstarsports.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 3.

6. Perform the necessary manual steps, as listed in the report. Because the fully qualified host name for the metadata server has changed, update the connection profiles for client applications where necessary.

7. Start the servers and perform the validation steps.

8. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Example 4: A Single-Machine Deployment Has Been Cloned to Another Machine

In this example, a single-machine deployment is installed on HOST01.example.com, and the deployment has been cloned to another machine called HOST99.example.com. In the new deployment, all references to the fully qualified host name, short host name, and IP address need to be updated to refer to the new host machine.
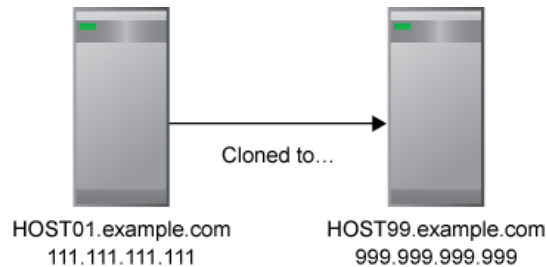
HOST01.example.com
111.111.111.111

Cloned to...

HOST99.example.com
999.999.999.999

In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Set up the new machine (HOST99.example.com) by using cloning, ghosting, or virtualization software. For details, see "Step 4: Change Existing Names at the Operating System Level, or Set Up New Machines to Be Identical to the Current Machines" on page 484.

3. On the new machine, use SAS Management Console to define a backup location for the new metadata server. Specify a different location than the one that is used by the old server.

4. Make a backup of the new metadata server.

5. Stop all of the servers on the new machine (HOST99.example.com), including the metadata server (or metadata server cluster). Stop the middle-tier processes on the old machine (HOST01.example.com).

6. Run the Update Host Name References tool on HOST99.example.com.

   In the Specify Host Names dialog box, you need to enter only the fully qualified host names. The DNS lookup identifies all other forms of the host names so that they can be updated correctly. Specify the following:

   • **Old host name: `HOST01.example.com`**

   • **New host name: `HOST99.example.com`**

   • **Perform DNS lookup**: Selected

7. Perform the necessary manual steps, as listed in the report. Because the metadata server host name has changed, update the connection profiles for client applications where necessary. If the deployment includes a middle tier, and if the new deployment will be running concurrently with the old deployment, then the multicast parameters might also need to be modified.

8. Start the servers and perform the validation steps.

9. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Example 5: A Multiple-Machine Deployment Has Been Cloned to a New Set of Machines

In this example, a multiple-machine deployment is installed on MET01.example.com, APP01.example.com, and MID01.example.com. Cloning software has been used to create a mirror of the deployment on a different set of machines called MET02.example.com, APP02.example.com, and MID02.example.com. In the new deployment, all occurrences of each machine's fully qualified name, short name, and IP address need to be updated to refer to the new machines.
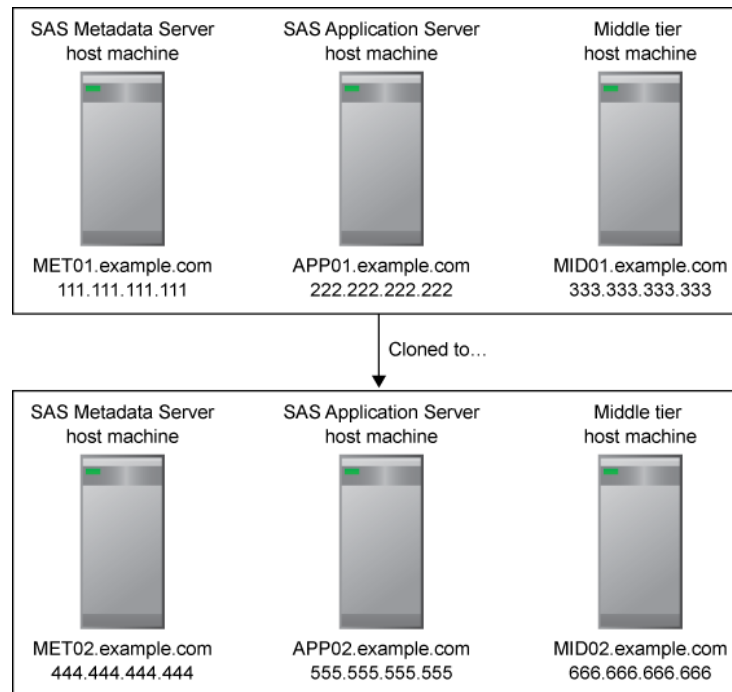
In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Set up the new machines (MET02.example.com, APP02.example.com, and MID02.example.com) by using cloning, ghosting, or virtualization software. For details, see "Step 4: Change Existing Names at the Operating System Level, or Set Up New Machines to Be Identical to the Current Machines" on page 484.

3. In the new deployment, use SAS Management Console to define a backup location for the new metadata server. Specify a different location than the one that is used by the old deployment.

4. Make a backup of the new metadata server.

5. Stop all of the servers in the new deployment, including the metadata server (or metadata server cluster). Stop the middle-tier processes in the old deployment.

6. Run the Update Host Name References tool on MET02.example.com.

   In the Specify Host Names dialog box, you need to enter only the fully qualified host names. The DNS lookup identifies all other forms of the old and new host names so that they can be updated correctly. Specify the following:

- **Old host name:**
  `MET01.example.com,APP01.example.com,MID01.example.com`
- **New host name:**
  `MET02.example.com,APP02.example.com,MID02.example.com`
- **Perform DNS lookup**: Selected

7. Run the Update Host Name References tool on APP02.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 4.

8. Run the Update Host Name References tool on MID02.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 4.

9. Perform the necessary manual steps, as listed in the report. Because the metadata server host name has changed, update the connection profiles for client applications where necessary. If the new deployment will be running concurrently with the old deployment, then the multicast parameters might also need to be modified.

10. Start the servers and perform the validation steps.

11. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Example 6: The Metadata Server Has Been Cloned to a New Machine

In this multiple-machine deployment example, cloning software has been used to move the metadata server from a machine called MET01.example.com to a new machine called MET99 example.com. All occurrences of the qualified name, short name, and IP address for the metadata server host machine need to be updated throughout the deployment to refer to the new host machine.
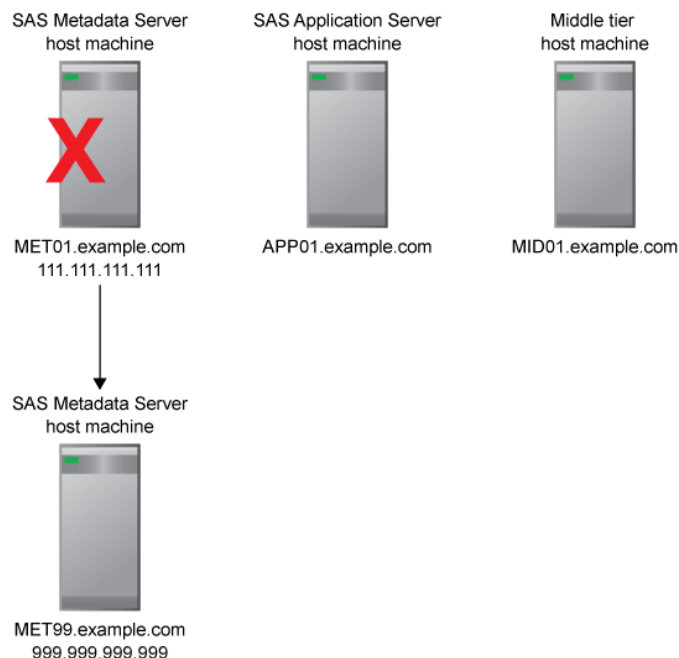
In this scenario, you would use the Update Host Name References tool as follows:

1. Make a complete backup of the current deployment, including a correct backup of the metadata server.

2. Set up the new machine (MET99.example.com) by using cloning, ghosting, or virtualization software. For details, see "Step 4: Change Existing Names at the Operating System Level, or Set Up New Machines to Be Identical to the Current Machines" on page 484.

3. Use SAS Management Console to define a backup location for the new metadata server. Specify a different location than the one that is used by the old server.

4. Make a backup of the new metadata server.

5. Stop the metadata server (or metadata server cluster) on the new machine (MET99.example.com). Stop all of the other servers in the deployment.

6. Run the Update Host Name References tool on MET99.example.com.

   In the Specify Host Names dialog box, you need to enter only the fully qualified host names. The DNS lookup identifies all other forms of the host names so that they can be updated correctly. Specify the following:

   - **Old host name: `MET01.example.com`**

   - **New host name: `MET99.example.com`**

   - **Perform DNS lookup**: Selected

7. Run the Update Host Name References tool on APP01.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 4.

8. Run the Update Host Name References tool on MID01.example.com. In the Specify Host Names dialog box, make exactly the same entries that you made in step 4.

9. Perform the necessary manual steps, as listed in the report. Because the metadata server host name has changed, update the connection profiles for client applications.

10. Start the servers and perform the validation steps.

11. Make a complete backup of the updated deployment, including a correct backup of the metadata server.

Creating a New Account for the Unrestricted User (Windows Only)

Refer to this topic if, in "Step 3: Check the Unrestricted User Account (Windows Only)" on page 484, you determined that the unrestricted user has only an external account that is qualified by a local machine name that is changing. In this situation, you must set up a new account to run the tool.

If the metadata server's machine name is changing, be sure to set up the new account before you change the machine name at the operating system level. If you are cloning the metadata server or the entire deployment, be sure to set up the account before you perform the cloning process.

Use either of the following procedures to set up an account:

- In the User Manager plug-in, add an internal account to the unrestricted user's definition. Use the internal account when you run the Update Host Name References tool.

- Or, in the User Manager plug-in, add another external account to the unrestricted user's definition. Qualify the account with the metadata server's new host name, and use this account when you run the Update Host Name References tool.

Note: If you rename the metadata server at the operating system level (or clone the metadata server to a new machine) before you have created the new account, it is still possible to connect to the server as an unrestricted user. You can do so by specifying the appropriate local account in the adminUsers.txt file, and then using that account to connect.

Chapter 34

Rebuild the Windows Service Definition

Rebuild Windows Service for Each SAS Web Application Server Instance

Note: This step is applicable to the February 15, 2022 and later releases of SAS 9.4M7 and SAS 9.4 M8 on Windows environments.

Prior to the February 15, 2022 release of SAS 9.4M7, SAS Web Application Server instances are managed with Pivotal tc Server. Thereafter, SAS Web Application Server instances are managed with Apache Tomcat. In releases where SAS Web Application Server instances are managed with Apache Tomcat, you must rebuild the Windows service definition for each instance of the SAS Web Application Server after updating the host name references or updating the wrapper.conf files. This is for Windows environments only.

SAS Web Application Server instances on Windows are managed with the SASServiceCtl.ps1 Windows PowerShell script. Rebuild the Windows service definition for each instance of the SAS Web Application Server with this script as follows:

1. Open a Windows command prompt from which you will run the commands in the following steps.

2. Navigate to the following directory that contains the SASServiceCtl.ps1 script:
   \SAS-configuration-directory\Lev*n*\Web\WebAppServer
   \SASServer*N_M*\bin

   Note: There is a copy of the SASServiceCtl.ps1 script in this directory for each SAS Web Application Server instance.

3. List each SAS Web Application Server instance.

   a. From Windows Services Manager, find the SAS Web Application Server instances, such as SASServer1_1 and SASServer2_1.

   b. Determine a common string that appears in the **Name** column for each SAS Web Application Server instance, for example, **WebAppServer**.

      Note: This string is used in the command to list all of the SAS Web Application Servers that contain that string in the service name.

c. Run the following command to list the server names of each SAS Web Application Server instance, substituting the common string that you identified in the last step as the value of the NAME option:

```
powershell -file .\SASServiceCtl.ps1 -action list -name
WebAppServer
```

d. Make a note of the **Service Name** and **Service Display Name** values for each SAS Web Application Server instance, as they are needed for subsequent commands.

4. Run the setenv.bat script to set the CATALINA_HOME environment variable, as follows: **setenv.bat**

   Note: The UNINSTALL option requires the existence of the CATALINA_HOME environment variable that defines where the core of the Tomcat software is installed.

5. Remove each instance of the SAS Web Application Server. For the value of the NAME parameter, use the name of the SAS Web Application Server. This name appears in the **Service Name** column from the resulting table in step 3 above.

```
powershell -file .\SASServiceCtl.ps1 -action uninstall
-name WebAppServer
```

For example, here is the command to uninstall the SAS Web Application Server instance named SAS-Config-Lev1-SASServer1_1-WebAppServer.

```
powershell -file .\SASServiceCtl.ps1 -action uninstall
-name SAS-Config-Lev1-SASServer1_1-WebAppServer
```

   Note: Run the command above for each instance of the SAS Web Application Server, substituting the correct name of the other SAS Web Application Server instances for the value of the NAME option..

6. Install the first SAS Web Application Server instance.

   a. Run the setenv.bat script to set the CATALINA_BASE environment variable for your first SAS Web Application Server instance.

      Note: The INSTALL option requires the existence of the following environment variables:

CATALINA_HOME	Defines where the core of the Apache Tomcat software is installed.
CATALINA_BASE	Defines the location where the server instance is to be installed.

   For example, suppose that the name of your first SAS Web Application Server instance is SAS-Config-Lev1-SASServer1_1-WebAppServer, and you are currently in the following directory for this instance: **\SAS-configuration-directory\Levn\Web\WebAppServer\SASServer1_1\bin**.

   Run the setenv.bat script from this directory as follows: **setenv.bat**

   b. Install the first SAS Web Application Server instance. For the value of the NAME parameter, use the name of the SAS Web Application Server. This name appears in the **Service Name** column from the resulting table in step 3 above.

```
powershell -file SASServiceCtl.ps1 -action install -name
WebAppServer
```

For example, here is the command to install the first instance of the SAS Web Application Server named SAS-Config-Lev1-SASServer1_1-WebAppServer. This command also includes the DISPLAY and DESCRIPTION options (with sample values) on the INSTALL command to specify the correct display name and description for that Windows service.

```
powershell -file .\SASServiceCtl.ps1 -action install -display
 "SAS [Config-Lev1] SASServer1_1 - WebAppServer" -description
"SASServer1_1 WebAppServer at Config-Lev1 on port 8080"-name
 SAS-Config-Lev1-SASServer1_1-WebAppServer
```

7. Install the other SAS Web Application Server instances.

   a. Run the setenv.bat script to set the CATALINA_BASE environment variable for your other SAS Web Application Server instances. The CATALINA_BASE environment variable must be set for each SAS Web Application Server instance. Therefore, you must run the script from the following directory for each SAS Web Application Server instance: **\SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer*N*_*M*\bin**

   For example, suppose that the name of your second SAS Web Application Server instance is SAS-Config-Lev1-SASServer2_1-WebAppServer, and that the directory for this instance is **\SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer2_1\bin**. Also, suppose that you are currently in the following directory for the first SAS Web Application Server instance: **\SAS-configuration-directory\Lev*n*\Web\WebAppServer\SASServer1_1\bin**.

   Run the setenv.bat script for your second SAS Web Application Server instance directly from the directory of the second SAS Web Application Server instance as follows:

   ```
   ..\..\SASServer2_1\bin\setenv.bat
   ```

   Note: In step c below, you are directed to repeat this step for the other SAS Web Application Server instances. Substitute the directory name of the other SAS Web Application Server instances for SASServer2_1.

   b. Install the other SAS Web Application Server instances:

   ```
   powershell -file SASServiceCtl.ps1 -action install -name
   WebAppServer
   ```

   For example, here is the command to install the instance of the SAS Web Application Server named SAS-Config-Lev1-SASServer2_1-WebAppServer. This command also includes the DISPLAY and DESCRIPTION options (with sample values) on the INSTALL command to specify the correct display name and description for that Windows service.

   ```
   powershell -file SASServiceCtl.ps1 -action install -display
   "SAS [Config-Lev1] SASServer2_1 - WebAppServer" -description
   "SASServer2_1 WebAppServer at Config-Lev1 on port 8180" -name
    SAS-Config-Lev1-SASServer2_1-WebAppServer
   ```

   c. Repeat steps a and b for the remaining SAS Web Application Server instances. Make sure to substitute the correct name of the other SAS Web Application Server instances.

Recommended Reading

Here is the recommended reading list for this title:

- *SAS Intelligence Platform: Overview*
- *SAS Intelligence Platform: Installation and Configuration Guide*
- *SAS Intelligence Platform: Migration Guide*
- *SAS Intelligence Platform: Security Administration Guide*
- *SAS Intelligence Platform: Application Server Administration Guide*
- *SAS Intelligence Platform: Desktop Application Administration Guide*
- *SAS Intelligence Platform: Data Administration Guide*
- *SAS Intelligence Platform: Middle-Tier Administration Guide*
- *SAS Intelligence Platform: Web Application Administration Guide*
- *SAS Environment Manager User's Guide*
- *Grid Computing in SAS*
- *SAS Companion for UNIX Environments*
- *SAS Companion for Windows*
- *SAS Companion for z/OS*
- *SAS Interface to Application Response Measurement (ARM): Reference*
- *SAS Language Interfaces to Metadata*
- *SAS Logging: Configuration and Programming Reference*
- *SAS System Options: Reference*
- *SAS/CONNECT User's Guide*
- *SAS/SHARE User's Guide*
- *Scheduling in SAS*
- SAS offers instructor-led training and self-paced e-learning courses to help you administer the SAS Intelligence Platform. For more information about the available courses, see support.sas.com/admintraining.

For a complete list of SAS publications, go to support.sas.com/en/books.html. If you have questions about which titles you need, please contact a SAS Representative:

SAS Books
SAS Campus Drive

Cary, NC 27513-2414
Phone: 1-800-727-0025
Fax: 1-919-677-4444
Email: sasbook@sas.com
Web address: support.sas.com/en/books.html

Glossary

access control template (ACT)
a reusable named authorization pattern that you can apply to multiple resources. An access control template consists of a list of users and groups and indicates, for each user or group, whether permissions are granted or denied.

ACT
See access control template.

aggregation
a summary of detail data that is stored with or referred to by a cube.

ancestor logger
a logger that is at a higher level in relation to other loggers in the logger hierarchy. For example, the Audit logger is an ancestor logger of Audit.Meta and Audit.Authentication.

appender
a named entity that represents a specific output destination for messages. Destinations include fixed files, rolling files, operating system facilities, and client applications.

Application Response Measurement (ARM)
the name of an application programming interface that was developed by an industry partnership and which is used to monitor the availability and performance of software applications. ARM monitors the application tasks that are important to a particular business.

ARM
See Application Response Measurement.

authentication
See client authentication.

authentication domain
a SAS internal category that pairs logins with the servers for which they are valid. For example, an Oracle server and the SAS copies of Oracle credentials might all be classified as belonging to an OracleAuth authentication domain.

authorization

the process of determining the permissions that particular users have for particular resources. Authorization either permits or denies a specific action on a specific resource, based on the user's identity and on group memberships.

autocall macro

a macro whose uncompiled source code and text are stored in an autocall macro library. Unlike a stored compiled macro, an autocall macro is compiled before execution the first time it is called.

base path

the location, relative to a WebDAV server's URL, in which packages are published and files are stored.

batch mode

a noninteractive method of running SAS programs by which a file (containing SAS statements along with any necessary operating system commands) is submitted to the batch queue of the operating environment for execution.

cache

a small, fast memory area that holds recently accessed data. The cache is designed to speed up subsequent access to the same data.

catalog

See SAS catalog.

channel

a virtual communication path for distributing information. In SAS, a channel is identified with a particular topic. Using the features of the Publishing Framework, authorized users or applications can publish digital content to the channel, and authorized users and applications can subscribe to the channel in order to receive the content.

client application

an application that runs on a client machine.

client authentication (authentication)

the process of verifying the identity of a person or process for security purposes.

client-side pooling

a configuration in which the client application maintains a collection of reusable workspace server processes.

connection profile

a client-side definition of where a metadata server is located. The definition includes a computer name and a port number. In addition, the connection profile can also contain user connection information.

conversion pattern

an expression that specifies an appender definition's pattern layout. A conversion pattern consists of a combination of user-supplied literal text and conversion specifiers.

credentials

the user ID and password for an account that exists in a particular authentication provider.

cube

    See OLAP cube.

custom repository

    an optional metadata store for a SAS Metadata Server that can be configured in addition to the foundation repository. Custom repositories are useful for physically segregating metadata for storage or security purposes.

data set

    See SAS data set.

DATA step

    in a SAS program, a group of statements that begins with a DATA statement and that ends with either a RUN statement, another DATA statement, a PROC statement, or the end of the job. The DATA step enables you to read raw data or other SAS data sets and to create SAS data sets.

database management system (DBMS)

    a software application that enables you to create and manipulate data that is stored in the form of databases.

DBMS

    See database management system.

default access control template (default ACT)

    the access control template (ACT) that controls access to a particular repository and to resources for which definitive access controls are not specified. You can designate one default ACT for each metadata repository. The default ACT is also called the repository ACT.

default ACT

    See default access control template.

descendant logger

    a logger that is at a lower level in relation to other loggers in the logger hierarchy. For example, Audit.Meta and Audit.Authentication are descendant loggers of the Audit logger.

development environment

    a computing environment in which application developers use software tools to write, compile, and debug programs.

diagnostic level (level)

    the degree of severity of a log event that can occur during SAS program processing. Examples of levels are trace, debug, informational, warning, error, and fatal.

domain

    a database of users that has been set up by an administrator by using a specific authentication provider such as LDAP or the host operating system. The domain name should be unique within your enterprise. For example, you should not have a Windows domain and a UNIX domain that are both named "SALES".

encryption

    the act or process of converting data to a form that is unintelligible except to the intended recipients.

external file
a file that is created and maintained by a host operating system or by another vendor's software application. An external file can read both data and stored SAS statements.

format
See SAS format.

foundation repository
the metadata repository that is used to specify metadata for global resources that can be shared by other repositories. For example, a foundation repository is used to store metadata that defines users and groups on the metadata server.

foundation services
See SAS Foundation Services.

generated transformation
in SAS Data Integration Studio, a transformation that is created with the Transformation Generator wizard, which helps you specify SAS code for the transformation.

grid computing
a type of computing in which large computing tasks are distributed among multiple computers on a network.

home directory (login directory)
the directory in which a user is placed after logging in. The home directory is also called the login directory.

identity
See metadata identity.

information map
a collection of data items and filters that provides a user-friendly view of a data source. When you use an information map to query data for business needs, you do not have to understand the structure of the underlying data source or know how to program in a query language.

Integrated Object Model (IOM)
the set of distributed object interfaces that make SAS software features available to client applications when SAS is executed as an object server.

Integrated Object Model server (IOM server)
a SAS object server that is launched in order to fulfill client requests for IOM services.

Integrated Windows authentication (IWA)
a Microsoft technology that facilitates use of authentication protocols such as Kerberos. In the SAS implementation, all participating components must be in the same Windows domain or in domains that trust each other.

internal account
a SAS account that you can create as part of a user definition. Internal accounts are intended for metadata administrators and some service identities; these accounts are not intended for regular users.

IOM

> *See* Integrated Object Model.

IOM server

> *See* Integrated Object Model server.

IWA

> *See* Integrated Windows authentication.

Java application

> a stand-alone program that is written in the Java programming language.

job

> a collection of SAS tasks that can create output.

level

> *See* diagnostic level.

library reference

> *See* libref.

libref (library reference)

> a SAS name that is associated with the location of a SAS library. For example, in the name MYLIB.MYFILE, MYLIB is the libref, and MYFILE is a file in the SAS library.

load balancing

> for IOM bridge connections, a program that runs in the object spawner and that uses an algorithm to distribute work across object server processes on the same or separate machines in a cluster.

locale

> a setting that reflects the language, local conventions, and culture for a geographic region. Local conventions can include specific formatting rules for paper sizes, dates, times, and numbers, and a currency symbol for the country or region. Some examples of locale values are French_Canada, Portuguese_Brazil, and Chinese_Singapore.

localhost

> the keyword that is used to specify the machine on which a program is executing. If a client specifies localhost as the server address, the client connects to a server that runs on the same machine.

log

> *See* SAS log.

logger

> a named entity that identifies a message category. Logger names have a hierarchical format that enables you to configure logging at a broad or a fine-grained level.

logging configuration

> an XML file or a set of SAS program statements that determines how log events are processed. You use the logging configuration to assign thresholds to loggers, to configure appenders, and to specify which categories and levels of log events are written to each appender.

logical server

the second-level object in the metadata for SAS servers. A logical server specifies one or more of a particular type of server component, such as one or more SAS Workspace Servers.

login

a SAS copy of information about an external account. Each login includes a user ID and belongs to one SAS user or group. Most logins do not include a password.

login directory

See home directory.

metadata

descriptive data about data that is stored and managed in a database, in order to facilitate access to captured and archived data for further use.

metadata identity (identity)

a metadata object that represents an individual user or a group of users in a SAS metadata environment. Each individual and group that accesses secured resources on a SAS Metadata Server should have a unique metadata identity within that server.

metadata model

a definition of the metadata for a set of objects. The model describes the attributes for each object, as well as the relationships between objects within the model.

metadata object

a set of attributes that describe a table, a server, a user, or another resource on a network. The specific attributes that a metadata object includes vary depending on which metadata model is being used.

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.

metadata server

a server that provides metadata management services to one or more client applications.

middle tier

in a SAS business intelligence system, the architectural layer in which web applications and related services execute. The middle tier receives user requests, applies business logic and business rules, interacts with processing servers and data servers, and returns information to users.

migrate

to populate a new deployment of SAS software with the content, data, or metadata (or a combination of these) from an existing deployment. Migrating might include upgrading to a new software release, converting data or metadata, or other changes to ensure compatibility.

migration

the process of moving data to a different server or storage device, converting the data for use in a different operating environment, or both. In a broader sense, the migration process can also include installing the newer software and converting applications for use with that software.

object spawner (spawner)

a program that instantiates object servers that are using an IOM bridge connection. The object spawner listens for incoming client requests for IOM services. When the spawner receives a request from a new client, it launches an instance of an IOM server to fulfill the request. Depending on which incoming TCP/IP port the request was made on, the spawner either invokes the administrator interface or processes a request for a UUID (Universal Unique Identifier).

OLAP

See online analytical processing.

OLAP cube (cube)

a logical set of data that is organized and structured in a hierarchical, multidimensional arrangement to enable quick analysis of data. A cube includes measures, and it can have numerous dimensions and levels of data.

OLAP schema

a container for OLAP cubes. A cube is assigned to an OLAP schema when it is created, and an OLAP schema is assigned to a SAS OLAP Server when the server is defined in the metadata. A SAS OLAP Server can access only the cubes that are in its assigned OLAP schema.

online analytical processing (OLAP)

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

planned deployment

a method of installing and configuring a SAS business intelligence system. This method requires a deployment plan that contains information about the different hosts that are included in the system and the software and SAS servers that are to be deployed on each host. The deployment plan then serves as input to the SAS Deployment Wizard.

plug-in

a file that modifies, enhances, or extends the capabilities of an application program. The application program must be designed to accept plug-ins, and the plug-ins must meet design criteria specified by the developers of the application program.

pool

a group of server connections that can be shared and reused by multiple client applications. A client-side pool consists of one or more puddles.

pooling

the act or process of creating a pool.

portlet

a web component that is managed by a web application and that is aggregated with other portlets to form a page within the application. Portlets can process requests from the user and generate dynamic content.

production environment

a computing environment in which previously tested and validated software is used (typically on a daily basis) by its intended consumers.

project repository
a metadata repository that serves as an individual work area or playpen. In general, each user who participates in change management has an individual project repository. (Project repositories are available for SAS Data Integration Studio only.)

promotion
the process of copying selected metadata and associated content within or between planned deployments of SAS software that could run different software releases. Methods of promotion include import and export processes, as well as explicit copies between two servers. This process is repeatable for a particular deployment.

publish
to deliver electronic information to one or more destinations. These destinations can include message queues, publication channels, and so on.

puddle
a group of servers that are started and run using the same login credentials. Each puddle can also allow a group of clients to access the servers.

resource template
an XML file that specifies the information that is needed for creating a metadata definition for a SAS resource.

role (user role)
a set of capabilities within an application that are targeted to a particular group of users.

root logger
the highest-level logger in the logger hierarchy. In a logging configuration, all other loggers inherit the root logger's attributes.

SAS Application Server
a logical entity that represents the SAS server tier, which in turn comprises servers that execute code for particular tasks and metadata objects.

SAS catalog (catalog)
a SAS file that stores many different kinds of information in smaller units called catalog entries. A single SAS catalog can contain different types of catalog entries.

SAS configuration directory
the location where configuration information for a SAS deployment is stored. The configuration directory contains configuration files, logs, scripts, repository files, and other items for the SAS software that is installed on the machine.

SAS Content Server
a server that stores digital content (such as documents, reports, and images) that is created and used by SAS client applications. To interact with the server, clients use WebDAV-based protocols for access, versioning, collaboration, security, and searching.

SAS data set (data set)
a file whose contents are in one of the native SAS file formats. There are two types of SAS data sets: SAS data files and SAS data views.

SAS Deployment Manager

a cross-platform utility that manages SAS deployments. The SAS Deployment Manager supports functions such as updating passwords for your SAS deployment, rebuilding SAS web applications, and removing configurations.

SAS Deployment Wizard

a cross-platform utility that installs and initially configures many SAS products. Using a SAS installation data file and, when appropriate, a deployment plan for its initial input, the wizard prompts the customer for other necessary input at the start of the session, so that there is no need to monitor the entire deployment.

SAS format (format)

a type of SAS language element that is used to write or display data values according to the data type: numeric, character, date, time, or timestamp.

SAS Foundation

the superset of all SAS software that is installable with the Base SAS installation. For a particular customer, the SAS Foundation is a collection of software consisting of the members of that superset required to support the deployment that the customer desires.

SAS Foundation Services (foundation services)

a set of core infrastructure services that programmers can use in developing distributed applications that are integrated with the SAS platform. These services provide basic underlying functions that are common to many applications. These functions include making client connections to SAS application servers, dynamic service discovery, user authentication, profile management, session context management, metadata and content repository access, information publishing, and stored process execution.

SAS Framework Data Server

a database server that is the default location for middle-tier data such as alerts, comments, and workflows, as well as data for the SAS Content Server and SAS Service Parts Optimization. The server is provided as an alternative to using a third-party DBMS. The server cannot be used as a general-purpose data store.

SAS IOM workspace (workspace)

in the IOM object hierarchy for a SAS Workspace Server, an object that represents a single session in SAS.

SAS library

one or more files that are defined, recognized, and accessible by SAS, and that are referenced and stored as a unit. Each file is a member of the library.

SAS log (log)

a file that contains a record of the SAS statements that you enter, as well as messages about the execution of your program.

SAS Management Console

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

SAS Metadata Model

a collection of metadata types that are used for saving information about application elements.

SAS Metadata Repository

a container for metadata that is managed by the 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.

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.

SAS Open Metadata Architecture

a general-purpose metadata management facility that provides metadata services to SAS applications. The SAS Open Metadata Architecture enables applications to exchange metadata, which makes it easier for these applications to work together.

SAS Open Metadata Interface

a set of methods that enable users to read metadata from or write metadata to the SAS Metadata Server.

SAS Pooled Workspace Server

a SAS Workspace Server that is configured to use server-side pooling. In this configuration, the SAS object spawner maintains a collection of workspace server processes that are available for clients.

SAS Stored Process (stored process)

a SAS program that is stored on a server and defined in metadata, and which can be executed by client applications.

SAS Stored Process Server

a SAS IOM server that is launched in order to fulfill client requests for SAS Stored Processes.

SAS system option (system option)

a type of SAS language element that is applied to any of a number of operations during a SAS session. System options can control SAS session initialization, SAS interactions with hardware and software, and input and output processing of SAS files.

SAS table

the visual rendering of a SAS data set in tabular format.

SAS Table Server

a data access server that supports the capabilities provided by SAS Shared Services for the SAS BI Dashboard, SAS Web Report Studio, and other SAS solutions.

SAS token authentication

a process in which the metadata server generates and verifies SAS identity tokens to provide single sign-on to other SAS servers. Each token is a single-use, proprietary software representation of an identity.

SAS Workspace Server
a SAS IOM server that is launched in order to fulfill client requests for IOM workspaces.

SAS/CONNECT spawner (spawner)
a program that runs on a remote computer and that listens for SAS/CONNECT client requests for connection to the remote computer. When the spawner program receives a request, it invokes a SAS session on the remote computer.

SAS/SHARE server
the result of an execution of the SERVER procedure, which is part of SAS/SHARE software. A server runs in a separate SAS session that services users' SAS sessions by controlling and executing input and output requests to one or more SAS libraries.

server tier
in a SAS business intelligence system, the tier in which the SAS servers execute. Examples of such servers are the SAS Metadata Server, the SAS Workspace Server, the SAS Pooled Workspace Server, the SAS Stored Process Server, and the SAS OLAP Server. These servers are typically accessed either by clients or by web applications that are running in the middle tier.

server-side pooling
a configuration in which a SAS object spawner maintains a collection of reusable workspace server processes that are available for clients. The usage of servers in this pool is governed by the authorization rules that are set on the servers in the SAS metadata.

source code repository
the location on the application server that contains stored process source code.

source metadata server
the metadata server from which metadata is promoted or replicated.

spawner
See SAS/CONNECT spawner.

spawner
See object spawner.

stored process
See SAS Stored Process.

subscriber
a recipient of information that is published to a SAS publication channel.

system option
See SAS system option.

target metadata server
the metadata server to which the metadata is promoted or replicated.

thread
the smallest unit of processing that can be scheduled by an operating system.

threshold

the lowest event level that is processed. Log events whose levels are below the threshold are ignored.

transformation

in data integration, an operation that extracts data, transforms data, or loads data into data stores.

trusted user

a privileged service account that can act on behalf of other users on a connection to the metadata server.

UNC

See Universal Naming Convention.

Universal Naming Convention (UNC)

a naming standard that is used to specify the file or directory path for networked computing resources such as servers, databases, and peripheral devices. UNC uses the format \\server-name\shared-resource-pathname.

unrestricted identity

a user or group that has all capabilities and permissions in the metadata environment due to membership in the META: Unrestricted Users Role (or listing in the adminUsers.txt file with a preceding asterisk).

user role

See role.

web application

an application that is accessed via a web browser over a network such as the Internet or an intranet. SAS web applications are Java Enterprise Edition (JEE) applications that are delivered via web application archive (WAR) files. The applications can depend on Java and non-Java web technologies.

Web Distributed Authoring and Versioning (WebDAV)

a set of extensions to the HTTP protocol that enables users to collaboratively edit and manage files on remote web servers.

WebDAV

See Web Distributed Authoring and Versioning.

WebDAV repository

a collection of files that are stored on a web server so that authorized users can access them.

workspace

See SAS IOM workspace.

Index