THE POWER TO KNOW.

SAS® Forecast Server 4.1
Administrator's Guide

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc 2011. *SAS® Forecast Server 4.1: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Forecast Server 4.1: Administrator's Guide

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

1st electronic book, July 2011
1st printing, July 2011

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

Contents

About This Book

Audience

The *SAS Forecast Server: Administrator's Guide* is for administrators who need to install, configure, and optimize the SAS Forecast Server. SAS and other programming expertise is not required.

For each audience, the required skill set is listed.

System Administrator
: The SAS Forecast Server uses the SAS Intelligence Platform. The system administrator should be familiar with the SAS Intelligence Platform. For more information, see **http://support.sas.com/documentation/onlinedoc/intellplatform**.

The system administrator performs these tasks:

- installation and configuration of the SAS Intelligence Platform and solution The system administrator should install and configure the required SAS Intelligence Platform software on the required operating system.

  To install the SAS Intelligence Platform on the Microsoft Windows operating system, the administrator should meet the following prerequisites:

  - be an administrator of the machine
  - be familiar with Windows concepts
  - know how to create folders
  - know how to run DOS BAT files
  - be familiar with Windows domain concepts in order to create user accounts and groups
- administration of the metadata for the SAS Forecast Server. The system administrator uses SAS Management Console to maintain the metadata for servers, users, and other global resources that are required by the SAS Forecast Server.
- if your solution uses ETL processes, administration of the metadata for SAS Data Integration Studio. The system administrator uses SAS Management Console to maintain the metadata for servers, users, and other global resources that are required by SAS Data Integration Studio.

Solution Administrator
: The solution administrator must perform these tasks:

- (optional) administration of the metadata for SAS Forecast Server. The solution administrator (or the system administrator) maintains the metadata for servers, users, and other global resources that are required by the SAS Forecast Server.
- maintain the solution's data and perform other administration tasks that enable users to analyze data.

Documentation Conventions

!SASROOT
represents the SAS Foundation directory.

SAS_CONFIG
represents the SAS configuration directory.

SAS_HOME
represents the SAS installation directory

Accessibility and Compatibility Features

The SAS Forecast Server includes accessibility and compatibility features that improve the usability of the product for users with disabilities, with some exceptions. These features are related to accessibility standards for electronic information technology that were adopted by the U.S. Government under Section 508 of the U.S. Rehabilitation Act of 1973. For more information about the accessibility of the SAS Forecast Server, see the online Help for SAS Forecast Studio or the *SAS Forecast Studio User's Guide*.

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

Recommended Reading

- administration documentation for the SAS Intelligence Platform at **http://support.sas.com/documentation/onlinedoc/intellplatform**.
- administration documentation for SAS Management Console at **http://support.sas.com/documentation/onlinedoc/sasmc**.
- *SAS High-Performance Forecasting User's Guide*
- *SAS/ETS User's Guide*

  SAS also publishes the following books, which are companions to the *SAS/ETS User's Guide* :

  - *SAS/ETS Software: Applications Guide 1, Version 6, First Edition* discusses features of SAS/ETS software for time series modeling and forecasting, financial reporting, and loan analysis.
  - *SAS/ETS Software: Applications Guide 2, Version 6, First Edition* discusses features of SAS/ETS software for econometric modeling and simulation.

For a complete list of SAS publications, go to support.sas.com/bookstore. If you have questions about which titles you need, please contact a SAS Publishing Sales Representative:

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-3228
Fax: 1-919-677-8166
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore

Chapter 1

SAS Forecast Server Overview

Understanding the Challenges in Forecasting

Businesses must understand the markets that they serve. To understand their markets, businesses must be able to analyze, model, and forecast the demand for their products and services. These products and services can be driven by many sales drivers, which might include input time series and calendar events. Business leaders must be able to interpret the results of these analyses and make decisions based on these results.

When the various products and services, categories, and geographies are considered, the number of time series, sales drivers, models, forecasts, and decisions can be tremendous. It is not uncommon that millions of time series must be modeled and forecast, and millions of decisions must be made based on these models and forecasts. Given the scale of the problem, customizing a time series model for each time series might not be feasible. An automated system that selects appropriate models and chooses influential sales drivers is required. The automated system must manage the time series data, time series models, calendar events, and results of the forecasting process in a scalable way. The results of the automated system must allow for "what-if" analysis, stochastic optimization, and goal-seeking support for making decisions.

Often businesses want to generate a large number of forecasts based on time-stamped data stored in their transactional or time series databases. Transactional databases

contain data from sources such as Web sites, point-of-sale (POS) systems, call centers, and inventory systems. A skilled analyst can forecast a single time series from such data by applying good judgment based on his or her knowledge and experience, by using various time series analysis techniques, and by using good software based on proven statistical theory. Generating frequent forecasts or large numbers of forecasts, however, requires some degree of automation.

Common forecasting problems that businesses face include the following:

- No skilled analyst is available.
- Many forecasts must be generated.
- Frequent forecast updates are required.
- Time-stamped data must be converted to time series data.
- Several sales drivers might, or might not, influence the time series.
- Several calendar events might, or might not, influence the time series.
- The forecasting model is not known for each time series.

What Is SAS Forecast Server?

SAS Forecast Server is an application that provides market-driven planning through accurate, dynamic demand forecasting and decision making.

Here are some applications of SAS Forecast Server:

- perform forecasting
- provide input to market response modeling applications
- provide input to time series data mining applications

The SAS Forecast Server provides automation and analytical sophistication to the forecasting process. By using SAS Forecast Server, which uses SAS High-Performance Forecasting, you can generate millions of forecasts in the turnaround time that is necessary to run your business. You can also uncover previously undetected trends, and you can predict future seasonal fluctuations. These capabilities create ample opportunities for you to reduce costs and increase revenues. (For more information about the procedures that are used, see "SAS High-Performance Forecasting" on page 8.)

Using SAS Forecast Server, you can do the following:

- produce trustworthy forecasts that reflect realities of your business
- focus your attention on the most critical forecasts by providing automatic, reliable forecasts on a large scale
- significantly reduce forecasting error
- improve inventory management
- improve forecasts for items that rarely sell

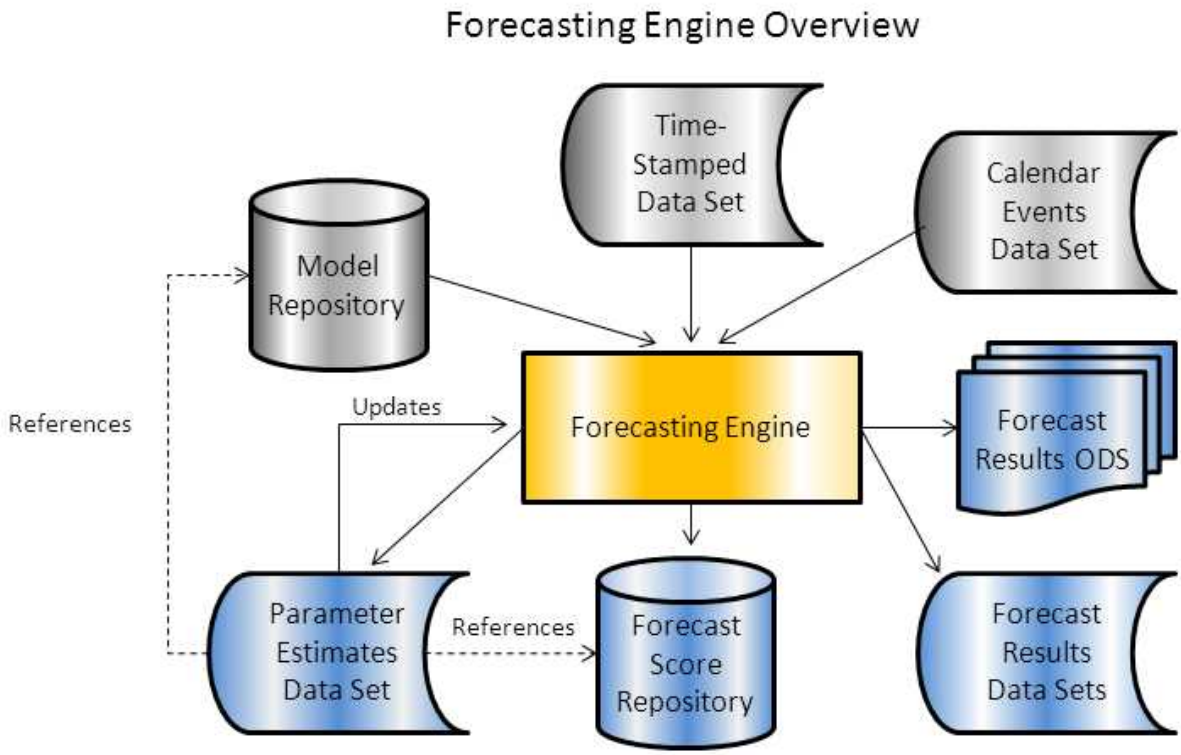
How SAS Forecast Server Works

Given a time-stamped data set, the software provides the following automatic forecasting process:

1. accumulates the time-stamped data to form a fixed-interval time series
2. diagnoses the time series using time series analysis techniques
3. creates a list of candidate model specifications based on the diagnostics
4. fits each candidate model specification to the time series
5. generates forecasts for each candidate fitted model
6. selects the most appropriate model specification based on either in-sample or holdout-sample evaluation using a model selection criterion
7. refits the selected model specification to the entire range of the time series
8. creates a forecast score from the selected fitted model
9. generates forecasts from the forecast score
10. evaluates the forecast using in-sample analysis and provides for out-of-sample analysis of forecast performance

The following figure shows the general flow of data in SAS Forecast Server.

Figure 1.1 *Data Flow in SAS Forecast Server*



When using SAS Forecast Server, you create or update forecasts by using the following general data flow:

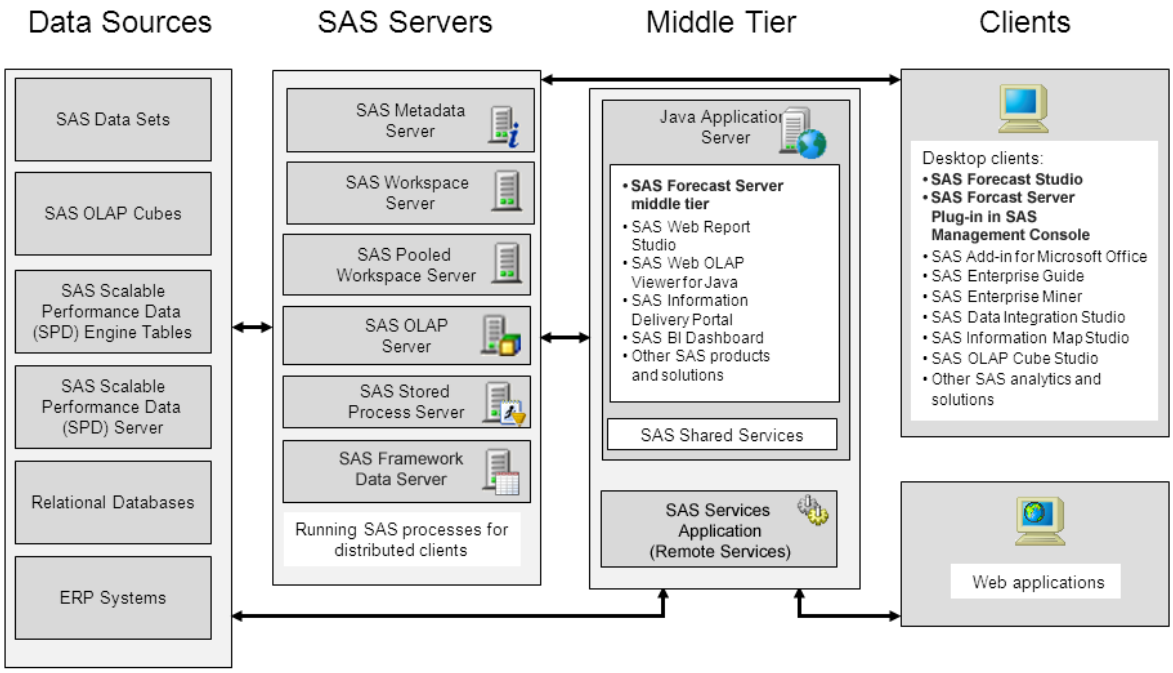
1. Create or generate an input SAS data set.
2. Open SAS Forecast Studio.
   a. Create a project.
   b. Select your input library and SAS data set.
   c. Assign variables to roles.
   d. (Optional) Configure the hierarchy.
   e. Set forecasting options.
   f. Finalize the project.
3. Create the forecasting model database.
4. Select the default model selection list.
5. Create events.
6. Generate forecasting results.
7. Modify estimates and forecast data again, if necessary, and repeat steps 3-6 (iterative process).
8. Store forecasting results and parameter estimates.

Overview of the SAS Forecast Server Architecture

Architecture Diagram

The following figure shows how SAS Forecast Studio and the SAS Forecast Server middle-tier pieces fit with the SAS Intelligence Platform.

Figure 1.2 *The SAS Forecast Server Architecture*



The SAS Intelligence Platform Components

The SAS Forecast Server uses the following components of the SAS Intelligence Platform:

SAS Metadata Server
: The SAS Metadata Server provides an enterprise-level repository for SAS server configurations and application management metadata. Products such as SAS Forecast Server store metadata concerning users and other resources. Administrators use SAS Management Console to administer the SAS Metadata Server, including SAS server configurations. The SAS Forecast Server uses the SAS Metadata Server to obtain metadata about SAS libraries and the SAS Workspace Server and SAS Object Spawner, and to authenticate users. A SAS System installation is required.

The SAS Workspace Server and SAS Object Spawner
: The SAS Workspace Server provides all computation and intermediate data storage services. A SAS System installation is required.

The SAS Forecast Server uses the SAS Workspace Server to execute SAS High-Performance Forecasting procedures and to save data to SAS data sets. The SAS Forecast Server accesses the SAS Workspace Server and the SAS Object Spawner through the middle tier.

SAS Web Infrastructure Platform
The SAS Analytics Platform has been deprecated, and SAS Forecast Server 4.1 now supports deployment to each of the application servers that are supported in SAS 9.3. Additional functionality that was previously provided by the SAS Analytics Platform is now provided by the SAS Web Infrastructure Platform. With the exception of the Log On dialog box in SAS Forecast Studio, these changes do not impact the user experience in SAS Forecast Studio, the primary client application for SAS Forecast Server.

SAS Services Application
The SAS Services Application deploys a set of services called Remote Services that are used by the SAS Information Delivery Portal, the SAS Stored Process Web application, and other Web applications. This application must be started before you start your Web application server.

The SAS Forecast Server Components

The SAS Forecast Server consists of the following components:

The SAS Forecast Server middle tier
The SAS Forecast Server middle tier is deployed onto an application server. The SAS Forecast Server middle tier is a collection of custom services that are used to support the SAS Forecast Studio clients and to coordinate their use of the underlying SAS Intelligence Platform.

SAS Forecast Studio
SAS Forecast Studio is the client application of SAS Forecast Server. With its graphical user interface, SAS Forecast Studio provides the user visual access to SAS Forecast Server projects. A middle tier connection is required by SAS Forecast Studio. The remote middle tier reports and manages the state of the project.

Note: All clients in SAS Forecast Server require this connection to the middle tier.

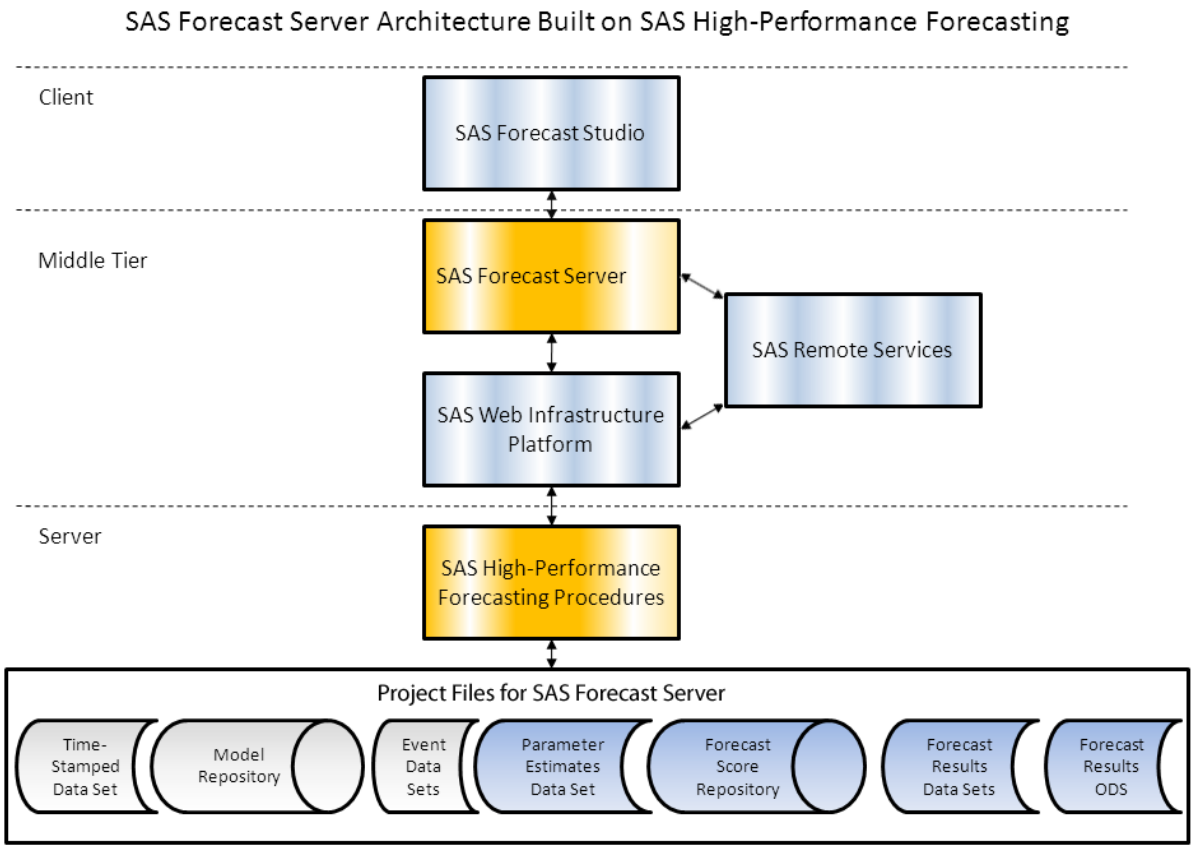
SAS Forecast Server Plug-in for SAS Management Console
The SAS Forecast Server Plug-in for SAS Management Console is a graphical user interface for managing the resources and content of the SAS Forecast Server middle tier. You can use the SAS Forecast Server Plug-in for SAS Management Console to perform the administrative tasks required to create and maintain an integrated environment. SAS Forecast Studio is used for the generation and use of content, and the plug-in is used to manage the product and its content. Examples of management actions include copying an environment, moving a project, converting a project to a new release, and deleting an environment.

The SAS Forecast Server Batch Interface
The batch interface consists of a collection of client macros for use within SAS Foundation. These macros send requests to the SAS Forecast Server middle tier to perform actions, including creating and managing projects and updating forecasts. Many of the actions parallel those that are also available through the SAS Forecast Server Plug-In for SAS Management Console.

The SAS Forecast Server Integration

The following figure shows the access points for SAS Forecast Studio, the SAS Forecast Server middle tier, and SAS High-Performance Forecasting components.

Figure 1.3 *The SAS Forecast Server Integration Architecture*



The following conditions must be true for SAS Forecast Server to run:

- The SAS Metadata Server is running.
- The SAS Object Spawner is running.
- SAS Remote Services are running.
- The SAS Web Infrastructure Platform and SAS Forecast Server are running on an application server.
- The input data for the SAS Forecast Server is accessible through configured SAS libraries.

SAS Forecast Studio accesses the SAS Forecast Server middle tier when it needs to access the SAS Metadata Server or the SAS Workspace Server. The SAS Forecast Server middle tier coordinates the use of the SAS Intelligence Platform (SAS Metadata Server and SAS Workspace Server) for the following purposes:

- access and return SAS library metadata from the SAS Metadata Server

- execute SAS High-Performance Forecasting procedures and return results
- authenticate users on the SAS Metadata Server

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

How SAS Forecast Server Relates to Other SAS Software

Overview

The SAS Forecast Server uses many of the procedures and options from other SAS products (such as SAS High-Performance Forecasting, SAS/ETS, Base SAS, and SAS/GRAPH). Many features not found in this solution software are available in other SAS solutions or in SAS products that are used with this SAS solution. If you do not find a feature that you need in this software, you might find it in one of the following SAS solutions or products.

SAS High-Performance Forecasting

SAS High-Performance Forecasting software provides a large-scale automatic forecasting system. The software provides for the automatic selection of time series models for use in forecasting time-stamped data.

The SAS Forecast Server uses the following SAS High-Performance Forecasting procedures that form the basis for the automatic forecasting capabilities:

HPFARIMASPEC
: The HPFARIMASPEC procedure is used to create an Autoregressive Integrated Moving Average (ARIMA) model specification file. The output of the procedure is an XML file that stores the intended ARIMA model specification. This XML specification file can be used to populate the model repository used by the HPFENGINE procedure. Likewise, you can also use the XML files generated by the other model specification procedures in this section to populate the model repository used by PROC HPFENGINE.

HPFDIAGNOSE
: The HPFDIAGNOSE procedure is an automatic modeling procedure to find the best model among ARIMA Models, Exponential Smoothing Models, and Unobserved Component Models.

  The HPFDIAGNOSE procedure has the following functionality:

  - intermittency test
  - functional transformation test
  - simple differencing and seasonal differencing test
  - tentative simple Autoregressive Moving Average (ARMA) order identification
  - tentative seasonal ARMA order identification
  - outlier detection
  - significance test of events

- transfer functions identification
- intermittent demand model
- exponential smoothing model
- unobserved component model

HPFENGINE
: The HPFENGINE procedure provides large-scale automatic forecasting of transactional or time series data. The HPFENGINE procedure extends the foundation built by PROC HPF, enabling the user to determine the list of models over which automatic selection is performed.

  The use of many forecast model families is supported when HPFENGINE is used in conjunction with new experimental procedures that generate generic model specifications. Among these models are the following:

  - ARIMA
  - Unobserved Component Models (UCM)
  - Exponential Smoothing Models (ESM)
  - Intermittent Demand Models (IDM)
  - External Models (EXM)

  Furthermore, users can completely customize the operation by defining their own code to generate forecasts.

  For models with inputs, the STOCHASTIC statement is especially helpful for automatically forecasting those inputs that have no future values.

  Also supported is the generation of a portable forecast score. The output of the SCORE statement is a file or catalog entry that you can use to efficiently generate forecasts outside of the HPFENGINE procedure when used with the HPFSCSUB function.

  The new HPFDIAGNOSE procedure produces output that is compatible with HPFENGINE. As a result, you can entirely automate the task of candidate model specification.

HPFESMSPEC
: The HPFESMSPEC procedure is used to create an Exponential Smoothing Model (ESM) specification file. The output of the procedure is an XML file that stores the intended ESM model specification.

HPFEVENTS
: The HPFEVENTS procedure provides a way to create and manage events associated with time series. The procedure can create events, read events from an events data set, write events to an events data set, and create dummies based on those events, if date information is provided.

  A SAS event is used to model any incident that disrupts the normal flow of the process that generated the time series. Examples of commonly used events include natural disasters, retail promotions, strikes, advertising campaigns, policy changes, and data recording errors.

  An event has a reference name, a date or dates associated with the event, and a set of qualifiers. The event exists separately from any time series. However, the event can be applied to one or more time series. When the event is applied to a time series, a dummy variable is generated that you can use it to analyze the impact of the event on the time series.

HPFEXMSPEC
: The HPFEXMSPEC procedure is used to create an External Model (EXM) specification file. The output of the procedure is an XML file that stores the intended EXM model specification.

HPFIDMSPEC
: The HPFIDMSPEC procedure is used to create an Intermittent Demand Model (IDM) specification file. The output of the procedure is an XML file that stores the intended IDM model specification.

HPFRECONCILE
: The HPFRECONCILE procedure reconciles forecasts of time series data at two different levels of aggregation. The procedure enables the user to specify the direction and the method of reconciliation, equality constraints, and bounds on the reconciled values at each point in time.

HPFSELECT
: The HPFSELECT procedure is used to create model selection lists. A model selection list contains references to candidate model specifications stored in the model repository. The output of the procedure is an XML file that stores the intended model selection list.

HPFUCMSPEC
: The HPFUCMSPEC procedure is used to create an Unobserved Component Model (UCM) specification file. The output of the procedure is an XML file that stores the intended UCM model specification.

For more information about SAS High-Performance Forecasting, see the *SAS High-Performance Forecasting User's Guide*.

SAS/ETS

SAS/ETS software provides SAS procedures that perform econometric and time series analysis and forecasting, as well as financial analysis and reporting. The software also provides an interactive environment for time series forecast and investment analysis. For more information about SAS/ETS software, refer to the *SAS/ETS User's Guide*.

SAS Enterprise Data Integration Server

SAS Enterprise Data Integration Server is an application that enables you to manage extract, transform, and load (ETL) process flows, which are sequences of steps for the extraction, transformation, and loading of data. SAS Enterprise Data Integration Server enables you to do the following:

- specify metadata for sources, such as tables in an operational system
- specify metadata for targets, such as tables and other data stores in a data warehouse
- create jobs that specify how data is extracted, transformed, and loaded from a source to a target

Base SAS

Base SAS delivers a highly flexible and extensible fourth-generation programming language that is specially designed for data access, transformation, and reporting. It includes a rich library of encapsulated programming procedures for data manipulation, information storage and retrieval, descriptive statistics, and report writing.

The output for the SAS Forecast Server is generated by the Output Delivery System that is part of Base SAS.

SAS/GRAPH

SAS/GRAPH software provides high-impact visuals for all levels of your organization, enabling customers to readily understand complex information and empowering them to make informed, timely decisions. SAS/GRAPH software extends the power of SAS data management, business intelligence, and analytic tools, enabling customers to turn data into full-color graphs and charts.

Chapter 2

Migration Tasks

Overview of the Migration Process

The tasks in this chapter explain the options that you have to migrate your content from previous releases of SAS Forecast Server to SAS Forecast Server 4.1. Evaluate the forecast plans and choose the time to migrate that best fits your business cycle.

Note: If you have projects created with SAS Forecast Server 1.4 and you want to use the SAS Migration Utility, then you must upgrade those projects to version 2.1 before upgrading to SAS Forecast Server 4.1.

These instructions explain the correct order of all migration tasks and point you to instructions in additional documents as necessary. It is recommended that you read this complete chapter before you begin the migration process for your SAS Forecast Server content.

For best results, it is recommended that you install the new SAS deployment on a separate group of machines from your original deployment.

About Project Organization

Prior to release 3.1, you did not have the ability to group projects together into workspaces. Releases 3.1 and beyond support forecasting environments to address this issue. A forecasting environment is defined by the following basic attributes:

- logical name
- host workspace server
- content path on server

A forecasting environment defines each of the following:

- a container for projects
- a workspace for users
- a partition of a workspace server

In releases 3.1 and beyond, forecasting environments replace workspace servers in the user interfaces of the product. However, the concept of forecasting environments also existed in earlier releases in a more basic form. In those releases, the product provided a single configurable "default location" value, which was effectively a globally shared "content path on server" value used for all workspace servers. This sharing had the effect of constraining all new projects to be stored at the same location in a workspace server file system (that is, the location referenced by the current "default location" value). Since release 3.1, you can use forecasting environments to identify multiple project locations for each workspace server, allowing for significantly more flexibility and control for site administrators. In addition, you can use metadata permissions to control access to each forecasting environment. Therefore, you can use forecasting environments to define private workspaces for groups of users, while still allowing sharing among group members.

Migration Considerations

Before migrating content from a previous release, review these issues:

- When updating a release prior to 3.1, the default location value was customizable. Since projects were not moved to the new location when the default location value was changed, this capability could result in a server having multiple areas for projects on the file system, even though new projects were created only in the current location. Review your existing projects carefully to determine the appropriate directories to migrate.

  Note: The SAS Migration Utility for SAS 9.1.3 provides an analyze mode that you can use to automatically detect existing project locations for that SAS release.

- Because forecasting environments support restricted access through the use of metadata permissions, you must determine appropriate access settings for each existing location prior to migration unless using an option that copies the permissions from the original deployment. In addition, support for project sharing provides another way for changing the access settings of a project within its forecasting environment.

- The user_locations.properties file is deprecated for releases 3.1 and later. The purpose of this file was to define a project area that was accessible to only a subset of the product users. This scenario is now supported by creating a forecasting environment definition and using the metadata permissions to make the forecasting environment accessible to only the specified subset of users.
- Starting with SAS Forecast Studio 3.1, sample reports are removed during migration. All sample reports use the FS_SAMPLE keyword. To ensure that no custom reports are removed during migration, custom reports should not use this keyword. All custom reports are migrated using the stored process support provided by the SAS Intelligence Platform framework.
- How SAS Forecast Server derives the owner for a project has changed from the 3.1 release. For more information, see "Project Owner Cannot Access Project" on page 167.

  CAUTION:
  Do not change the metadata name in SAS Management Console.
- One of the post-configuration steps in SAS Forecast Server 2.1 was to create an HPF server definition in SAS Management Console. When you migrate from SAS Forecast Server 2.1 to SAS Forecast Server 4.1, the HPF server definition remains configured under the Server Manager in SAS Management Console 9.3. After you have configured SAS Forecast Server 4.1, you should remove the HPF server definition in SAS Management Console before attempting to access any of the SAS Forecast Studio tasks in SAS Enterprise Guide or the SAS Add-In for Microsoft Office.

What Happens during Migration?

Primarily, migration involves copying the forecasting environment and project details to the new deployment. How exactly this is accomplished depends on the source deployment.

When migrating from release 3.1 and beyond, the source deployment will contain compatible metadata that describes the forecasting environments and projects. In this case, migration consists of copying that metadata to the new deployment.

For older releases, no forecasting-environment metadata is available in the source deployment. However, since the relative path to each project is consistent and known, the implied location of the environment in the file system can be derived from the full path to the project files. This path is recorded in the project metadata, and allows definitions for forecasting environments to be automatically generated during migration.

In either case, because of the potential size of the file system content, most migration options focus on moving only the metadata that describes the content. As a result, you must manually move the file system content to complete the migration when using either of these strategies.

When manual content movement is necessary, the files on the destination machine must have the same file paths as they had on the source machine to ensure project integrity. For example, you must place content located at `C:\Forecasting` at the same location on the destination system. Neither the drive letter nor the directory names can change. This is required because the file path is stored at several locations within the project content. If you want to change the file paths during your upgrade, see "Option 3: Archives" on page 19.

Note: If you are migrating content on the same machine, see "Migrating Environments on the Same Machine" on page 21.

Although you have migrated these projects to the new deployment, these projects are still formatted for the previous release of SAS Forecast Server. You must update the project version so that the internal structure of the projects is updated to use the formatting in the current release of SAS Forecast Server. If you have permission to update the project version, then SAS Forecast Studio automatically prompts you to convert the project when you open it for the first time. You can also convert these projects manually by using the SAS Forecast Server macros or the SAS Forecast Server Plug-in for SAS Management Console.

Note: Before migrating any project, create a backup for the project. You can use this backup copy to restore your project if there are any problems encountered during the migration or when you update the project version.

About Migrating Archives

As with all file content, you must manually migrate project archives. Since no search is performed for these files, no reminders or warnings will be generated for them during the migration process.

Migration Options

Overview of the Migration Options

When migrating from a previous release of SAS Forecast Server, you can choose from three migration options.

1. The SAS Migration Utility. Use this option when you need to perform a system migration. The migrated path for the SAS Forecast Server environment must be in the same location as the source system.
2. Automatic Registration. Use this option when you are migrating only SAS Forecast Server. The migrated path for the SAS Forecast Server environment must be in the same location as the source system.
3. Archives. Use this option to change the path of the SAS Forecast Server projects during the upgrade to the new release.

Note: If you are migrating content on the same machine, see "Migrating Environments on the Same Machine" on page 21.

Option 1: The SAS Migration Utility

About Migration Using the SAS Migration Utility

The standard migration mechanism for all SAS products is the SAS Migration Utility. This application generates a package of migrated content and settings that the SAS Deployment Wizard uses to configure the new system. This option focuses on migration of the entire system. The SAS Migration Utility does not package the file system content

for SAS Forecast Server due to the expected size. Instead, you are required to manually move the content for SAS Forecast Server to the corresponding locations on the new system. However, the application does analyze and migrate the metadata for the content. The application also issues warnings to draw your attention to special concerns, which might require consideration or additional planning.

Note: For content that you are migrating from SAS Forecast Server 2.1, it is recommended that you use the analyze mode provided by the SAS Migration Utility for SAS 9.1.3. It is recommended that you do this even if you do not intend to use this utility to migrate your data. This approach is recommended because the SAS Migration Utility can assess the state of your source system. The report that it generates identifies all projects and forecasting environments that you should migrate. Using this report reduces the chance of losing content during migration.

About Project Description and Version Information

The SAS Migration Utility does not have access to the project description and version information in releases prior to 3.1. During migration, default values are provided in the metadata. The SAS Migration Utility assigns **<3.1** for the version information, and the project description is blank. However, after you use the migrated project, the description for it is automatically restored in the metadata.

Alternatively, you can unregister a forecasting environment and re-register it in the SAS Forecast Server Plug-in for SAS Management Console to recreate the metadata. For more information about using these commands, see the Help for the SAS Forecast Server Plug-in available in SAS Management Console.

How to Migrate with the SAS Migration Utility

To perform a migration with the SAS Migration Utility:

1. Follow the instructions in the *SAS 9.3 Intelligence Platform: Migration Guide* located at `http://support.sas.com/documentation/onlinedoc/intellplatform` to run the SAS Migration Utility executable.
2. Review the analysis report generated by the SAS Migration Utility.

   TIP If you are migrating from release 2.1, the analysis report generated by the SAS Migration Utility will list the file system paths for all implied forecasting environments. Those locations are detected indirectly using the project definitions. Therefore, only locations that host at least one active project are detected.
3. For each of your forecasting environments, copy your content files from the source machines to the corresponding locations on the new destination machines.

   Note: You must create a directory structure on your new system that mirrors the directory structure of your original system. For example, if the content was located in the `C:\Forecasting` directory, then you must also create a `C:\Forecasting` directory on the destination machine to store those files in the new system.
4. Start the SAS Deployment Wizard executable from your SAS Software Depot.
5. When prompted, choose to **Perform migration**, and then click **Browse** to specify the **SAS Migration Utility Package Directory**.
6. Complete the remaining steps in the SAS Deployment Wizard.
7. Assign appropriate user permissions to the forecasting environment directories on the new system.

8. Review and configure metadata permissions as desired. The metadata that is copied from the source deployment retains its permissions. However, any metadata that is generated is configured to use default permissions.

Note: You can use the update batch operation to update multiple projects to use the current version. In the SAS Forecast Server Plug-in for SAS Management Console, right-click on the forecasting environment node, and select **Batch Operations** ⇨ **Update**. You can also perform an update for single projects. For more information about the Update operations, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Option 2: Automatic Registration

About Migration Using Automatic Registration

The SAS Forecast Server Plug-in for SAS Management Console includes several forecasting environment-level operations that you can use for migration. The automatic registration feature provided in the SAS Forecast Server Plug-in for SAS Management Console searches a newly registered forecasting environment for any unregistered projects and automatically registers them. Using this feature, you can manually create a new forecasting environment definition (using the **Register Environment** command) that points to a directory structure copied from an old installation of SAS Forecast Server. In addition, this feature enables you to select the automatic registration option to re-create the metadata registrations for all content in the structure.

Note: Using this method, you must still copy the file system content to the same location in the new system to ensure project integrity.

How to Migrate Using Automatic Registration

To migrate using automatic registration:

1. For each of your forecasting environments, copy your content files from the source machines to the corresponding locations on the new destination machines.

   Note: You must create a directory structure on your new system that mirrors the directory structure of your original system. For example, if the content was located in the `C:\Forecasting` directory, then you must also create a `C:\Forecasting` directory on the destination machine to store those files in the new system.

   TIP If you are migrating from release 2.1, the analysis report generated by the SAS Migration Utility will list the file system paths for all implied forecasting environments. Those locations are detected indirectly using the project definitions. Therefore, only locations that host at least one active project are detected.

2. Assign appropriate user permissions to the forecasting environment directories on the new system.

3. After you have installed and configured the product, open SAS Management Console, and log on with a user account that has been configured to access the SAS Forecast Server Plug-in for SAS Management Console. This user must also be configured to use the SAS Forecast Server administrator capabilities. For more information, see "How to Configure the SAS Forecast Server Users" on page 44.

4. Select the **Forecast Server** node in the **Plug-ins** tab.

5. Right-click on the **Forecast Server** node, and select **Register Environment**. The Register Environment dialog box appears.
6. In the resulting dialog box, do the following:
   a. Type the name that you would like to assign to the forecasting environment.
   b. (Optional) Type a description for the forecasting environment.
   c. Select the **Host name** of the SAS Workspace Server. The first available workspace server is selected by default. You can select a different server using the drop-down list.
   d. Type the location or click **Browse** to specify the location of the file system content.
   e. Specify the metadata location of the **Reports folder**.
   f. Select **Automatically register all projects within this environment** to execute project registration.
   g. Click **OK**.
7. Assign appropriate user permissions to the forecasting environment directories on the new system.
8. Review and configure metadata permissions as desired on the project metadata objects.

Note: If you have an existing forecasting environment or if you opted to not automatically register all of the projects in the forecasting environment using the Register Environment dialog box, you can use the register batch operation to register multiple projects at any time. Right-click on the forecasting environment node, and select **Batch Operations** ⇨ **Register**. You can also use the update batch operation to update multiple projects to use the current version. In the SAS Forecast Server Plug-in for SAS Management Console, right-click on the forecasting environment node, and select **Batch Operations** ⇨ **Update**. You can also perform an update for single projects. For more information about the update or batch operations, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Option 3: Archives

About Migration Using Archives

As in previous releases, you can archive individual projects into a portable format and then unarchive them to working projects in a separate installation. This functionality is supported through the macro interfaces, including the bulk archiving operations, and also through the SAS Forecast Server Plug-in for SAS Management Console. While slower, the use of archives supports considerably more cases than the previous two options. For example, migration across architectures, migration across forecasting environments, selective migration, and so on. Unlike the previous options, this option also moves the file system content within the archive files. However, archive-based migration is limited to migrating projects. Other means must be used to migrate the forecasting environment details.

How to Migrate Using Archives

To migrate using archives:

1. Export the project created with the previous version of SAS Forecast Server.

- If you are upgrading from SAS Forecast Server 2.1, you must use the FSEXPORT macro in SAS Forecast Server 3.1 to export the project. For more information, see the FSEXPORT and FSEXPALL macros in the *SAS Forecast Server 3.1: Administrator's Guide*.
- If you are upgrading from SAS Forecast Server 3.1 or later, you can use the FSEXPORT and FSEXPALL macros of the SAS Forecast Server Plug-in for SAS Management Console to archive your projects.

2. After you have installed and configured the product, open SAS Management Console, and log on with a user account that has been configured to access the SAS Forecast Server Plug-in for SAS Management Console. This user must also be configured to use the SAS Forecast Server administrator capabilities.
3. Select the **Forecast Server** node in the **Plug-ins** tab.
4. Right-click on the **Forecast Server** node, and select **New Environment**. The New Environment dialog box appears.
5. In the New Environment dialog box, complete these steps:
   a. Type the name that you would like to assign to the forecasting environment.
   b. (Optional) Type a description for the forecasting environment.
   c. Select the **Host name** of the SAS Workspace Server. The first available workspace server is selected by default. You can select a different server using the drop-down list.
   d. Type the location or click **Browse** to specify the location of the file system content.
   e. Specify the metadata location of the **Reports folder**.
   f. Click **OK**.
6. Assign appropriate user permissions to the forecasting environment directories on the new system.
7. Right-click on the forecasting environment node where you want to add the project, and then select **Unarchive Project**. The Unarchive dialog box appears.
8. Type the location for the archive file, or click **Browse** to specify the location of the archive file. Click **OK**.

Note: Alternatively, you can use the unarchive batch operation to import multiple projects when the archive files are all in the same directory. Right-click on the forecasting environment node, and select **Batch Operations ⇨ Unarchive**. You can also use the update batch operation to update multiple projects to use the current version. In the SAS Forecast Server Plug-in for SAS Management Console, right-click on the forecasting environment node, and select **Batch Operations ⇨ Update**. Updating single projects is also supported. For more information about the update or batch operations, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Migrating Environments on the Same Machine

About Performing a Migration on the Same Machine

When you use the SAS Migration Utility and the SAS Deployment Wizard to perform a migration on the same machine, the destination (also called migrated) deployment is staged with a copy of the product metadata from the source deployment. However, the copied metadata references the same file system content as the source deployment because the metadata configuration is intended for cases where the destination deployment replaces the source deployment. When this replacement occurs, the destination deployment inherits the file system content.

If you intend to continue using the source deployment in addition to the destination deployment, then additional steps are required. You must create a copy of the content on the file system in a different location, and you must change the metadata in the destination deployment to reference the copied content rather than the content that belongs to the source deployment. To ensure that all metadata references and internal references are updated appropriately, the copies must be made using SAS Forecast Server. Do not move file content using the tools in your operating system because doing so corrupts the migrated content.

How to Migrate Environments on the Same Machine

Note: Complete these steps after using the SAS Migration Utility.

To create two independent deployments of SAS Forecast Server that use the same file system:

1. Shut down the source deployment of SAS Forecast Server.

   CAUTION:
   Running two deployments of SAS Forecast Server simultaneously when both deployments access the same content areas can result in data loss. By shutting down the source deployment, you are giving the destination deployment exclusive access to the content, so that the content can be copied. Each deployment will have its own copy of the content.

2. For each migrated environment, use the tools from the destination deployment to clone the content from the source deployment. For more information, see "Cloning a Forecasting Environment" on page 97.

3. Use either the SAS Forecast Server Plug-in for SAS Management Console or the SAS Forecast Server macros to unregister each of the migrated environments in the previous step. Only the cloned environments should remain.

   CAUTION:
   Do not use the Delete action instead of the Unregister action. The Unregister action updates the metadata of the environment without changing the content in the file system. The Delete action removes content from the file system.

4. (Optional) Use either the SAS Forecast Server Plug-in for SAS Management Console or the SAS Forecast Server macros to rename each cloned environment to match the name of the environment from which it was cloned.

For example, the environment that you copied in step 2 was called Sales, so you named the copied environment SalesCopy. Now that only the SalesCopy environment exists, you can rename this environment Sales, and there will be no naming conflict.

Now that each deployment has its own exclusive area for environments, you can restart SAS Forecast Server in the source deployment.

Migrating Reports

Directories Specified in the Source File Field

An incompatibility was identified in the 2.1 release between the configuration of the sample report objects and the SPK files. Specifically, the **Source file** field for a stored process metadata object is intended to store only a simple filename (that is, it should not include a directory name). The sample reports available in this release correct this issue, but it might be present in any custom reports that you migrate from the 2.1 release. When importing an SPK file that contains stored process objects in this state, the attempt to write the source file to the file system fails unless you manually create the directories included in the **Source file** field of the stored process metadata object beforehand.

To successfully migrate stored process metadata objects in this state, you should do one of the following:

- Correct the value of the **Source file** field in the old system before generating the SPK file.
- Manually create the directories in the target system before importing the contents of the SPK file. If you use this option, it is recommended that you also correct the field values in the target system to avoid future problems.

Report Migration Considerations

Before migrating your reports from SAS Forecast Server 2.1, you must consider how you would like to organize them in the new system. Prior to the 3.1 release, reports for all projects were stored together in a single global tree in metadata, which was shown without filtering in the Reports and Stored Processes dialog box in SAS Forecast Studio. It was the user's responsibility to ensure that the stored process was indeed a report and that it was configured to execute on the workspace server hosting the project files.

Starting with release 3.1, you can have a separate reports area within metadata for each forecasting environment. In addition, the location of this area is now configurable (using the SAS Forecast Server Plug-in for SAS Management Console). When a report is registered in metadata, its stored process metadata object is configured to use the FS_REPORT keyword. The sample reports are configured to use the FS_SAMPLE keyword. In addition, the reports that appear in the Reports and Stored Processes dialog box are filtered to avoid showing any reports that are not appropriate for the current user and project.

Note: Because the reports are managed the same way in the 3.1 and 4.1 releases, you do not need to reorganize your reports when migrating from 3.1 to 4.1.

To appear in the SAS Forecast Studio Reports and Stored Processes dialog box, your report must meet the following conditions:

- The reports area for the project's forecasting environment must be configured. For more information, see "Managing Reports" on page 77.
- The report's stored process metadata object must be located in the configured reports area. For more information, see "Registering Reports" on page 80.
- The user must have the ReadMetadata permission for the report object.
- The report's stored process metadata object must have the special keyword *FS_REPORT*, which identifies it as a SAS Forecast Server report.
- The report's stored process metadata object must be configured to execute on the same server that hosts the project (that is, the host server of the project's forecasting environment).

Note: To simplify the task of migration of older reports that lack the FS_REPORT keyword, the SAS Forecast Server Plug-in for SAS Management Console provides a **Search** command. This command automatically adds the *FS_REPORT* keyword to any stored process found in the configured reports area. It is recommended that you use this command when importing a large collection of reports.

Due to the above considerations, your report migration might require that you package the set of reports in multiple SPK files so that you can import them into separate report areas. For more information about configuring reports, see Chapter 10, "Report Administration Tasks," on page 75. For more information about the SAS Forecast Server Plug-in for SAS Management Console, see the Help in SAS Management Console.

Import Reports to SAS Forecast Server 4.1

To import reports to SAS Forecast Server 4.1:

1. Open SAS Management Console, and log on as the SAS Administrator (for example, sasadm@saspw).
2. Select the **Folders** tab.
3. Right-click on your root report folder, and select **Import SAS Package**. The Import SAS Package Wizard opens.
4. Type the location to the SPK file, or click **Browse**. Choose to import all objects. Click **Next**.
5. In the Select Objects to Import step, all objects are selected for import. You can select a subset of the files to import if you prefer not to import all of them. Click **Next**.
6. In the About Metadata Connections step, information about the collection of objects that you are importing, application servers, and other servers appears. Click **Next**.
7. In the SAS Application Servers step, specify the **SAS Application Server**. Click **Next**.
8. In the Source Code Repositories step, select the target path for the source code repository. Click **Next**.
9. Review the Summary page, and then click **Next** to start the import process.
10. After the import process completes, click **Finish**.

Post-Migration Tasks for Reports

Note: These steps are required only when migrating from SAS Forecast Server 2.1.

After you have migrated reports from SAS Forecast Server 2.1 to the current release, you might also need to make code modifications. Assess such modifications on a case-by-case basis based on the following factors:

- changes to the macros provided for reports, including some new macros
- a change to the manner in which multiple-value parameters are passed to the code
- enhanced support for libraries
- enhanced support for dynamically populated parameters (for example, parameters with options that are generated by the application at run time)
- enhanced output format and style support

Note: For more information about making code modifications based on the above factors, see "Creating Reports" on page 82.

In most cases, to make your reports usable with this release of SAS Forecast Server, you can update the preamble code in the source code file to use the same code that is used by the sample reports. The preamble code used by the sample reports is shown below.

```
*ProcessBody;

/*---------------------------------------------------------*/
/*- initialize the HPF stored process support             -*/
/*---------------------------------------------------------*/
%hpfstp();

/*---------------------------------------------------------*/
/*- initialize the ODS output                             -*/
/*---------------------------------------------------------*/
%HPF_InitODSOutput();

%stpbegin;

/*---------------------------------------------------------*/
/*- convert any list parameters to space-delimited lists  -*/
/*---------------------------------------------------------*/
%HPF_MergePromptListParameters();

/*---------------------------------------------------------*/
/*- include the LIBNAME statements for the project        -*/
/*---------------------------------------------------------*/
%include "&HPF_INCLUDE";
```

Chapter 3

Pre-Installation Tasks

Verify System Requirements

Review the system requirements documentation before you install SAS Forecast Server, to ensure that your system meets the appropriate requirements. For more information, see **http://support.sas.com/documentation/installcenter/en/ikorvarofrsr/64683/PDF/default/sreq.pdf**

Note: You must have a license for SAS/OR to access its full functionality. The SAS Forecast Server has internal access to limited SAS/OR functionality, but it does not offer a direct interface to SAS/OR.

CAUTION:

You must use version 4.1 of both SAS Forecast Server and SAS High-Performance Forecasting. 4.1 is the version available in your software bundle. If you use version 4.1 of SAS Forecast Server with an earlier version of SAS High-Performance Forecasting, then you might get unexpected results.

Review Requirements for the Input Data Set

For SAS Forecast Server to generate a forecast, the input data set must meet these requirements:

- one variable for each time series
- a date or datetime variable to generate forecasts
- a single SAS data set that you define in a SAS library.

For more information about how to prepare your input data, see the *SAS Forecast Studio: User's Guide*.

Migrate Content to This Release of SAS Forecast Server

If you are upgrading from a previous release, review the following information:

Migration Tasks for SAS Forecast Server
: This chapter provides instructions about how to migrate your content from a previous release of SAS Forecast Server when you upgrade to SAS Forecast Server 4.1. For more information, see Chapter 2, "Migration Tasks," on page 13.

SAS 9.3 Intelligence Platform: Migration Guide
: This document contains migration information that is specific to the SAS Intelligence Platform. It covers topics ranging from how to plan for a migration to how to validate the migrated SAS deployment of the SAS Intelligence Platform. To access this guide, see **http://support.sas.com/documentation/onlinedoc/intellplatform**.

Complete the Pre-installation Tasks for the SAS Intelligence Platform

Before you begin to install the SAS Intelligence Platform and SAS Forecast Server, you must complete a set of pre-installation tasks for the SAS Intelligence Platform. These tasks include installing various third-party components, confirming your operating system requirements, creating the required user accounts, and obtaining your SAS software.

For information, see the *SAS Intelligence Platform: Installation and Configuration Guide* at **http://support.sas.com/documentation/onlinedoc/intellplatform**.

Create an Operating System Account for Product Administrators and Users

About the User Accounts for SAS Forecast Server

SAS Forecast Server uses two types of user accounts:

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

  You must create the operating system account for the administrator as a pre-installation task. For more information, see "Create an Operating System Account for the Product Administrator" on page 27.

Users of SAS Forecast Studio
: These users must have a valid host operating system account, and you must associate that account with a metadata user using SAS Management Console. In addition, these users must have full access to the appropriate project areas on each SAS Workspace Server.

  You can create regular user accounts for SAS Forecast Server as a post-installation task. For more information, see "Managing Roles and Capabilities" on page 39.

Create an Operating System Account for the Product Administrator

On each SAS Workspace Server that has SAS Forecast Server projects, create an operating system account for the administrator of SAS Forecast Server.

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

- If you are working on a local machine, open the Windows User Manager and complete these tasks:
  - Clear the **User must change password at next logon** check box.
  - Select the **User cannot change password** check box.
  - Select the **Password never expires** check box.
  - Grant the user permission **Log on as a Batch Job**.

  Note: In the Windows user manager, you cannot enter <*domain*>*username* (you enter the user name only), but you must enter <*domain*>*username* in the SAS Deployment Wizard and SAS Management Console.
- Define the user (for example, <*domain*>*username* on the Active Directory server.

Create Operating System Accounts for Users of SAS Forecast Studio

You can create the SAS Forecast Server UNIX user group as a pre-installation or post-installation task. For more information, see "Creating Operating System Accounts in UNIX Environments" on page 37.

Obtain a Deployment Plan

Before you can install SAS Forecast Server, you must obtain a deployment plan. The deployment plan is a summary of the software that is installed and configured during your installation. A deployment plan file, named plan.xml, contains information about what software should be installed and configured on each machine in your environment. This plan serves as input to the SAS installation and configuration tools.

SAS includes a standard deployment plan. You can use this standard plan or create your own plan.

For more information, see "About Deployment Plans" in the *SAS Intelligence Platform: Installation and Configuration Guide* at **http://support.sas.com/documentation/onlinedoc/intellplatform**.

Create a SAS Software Depot

Download the software that is listed in your SAS Software Order with the SAS Download Manager. This creates a SAS Software Depot, which includes the SAS installation data (SID) file. The SID file is used by the SAS system to install and license SAS software. It is a control file that contains license information that is required to install SAS. After you have downloaded the SAS Software Depot, you can then use the SAS Deployment Wizard to install your software. For more information, see "Creating a SAS Software Depot" in the *SAS Intelligence Platform: Installation and Configuration Guide* at **http://support.sas.com/documentation/onlinedoc/intellplatform**.

Determine the Location of the SAS Environment URL

During deployment, you are prompted by the SAS Deployment Wizard to specify a URL location of the SAS environment file, named sas-environment.xml (for example, **http://<your HTTP server>/sas-environment.xml**). This file defines a set of SAS deployments at your site for client applications to use. The sas-environment.xml file does not need to physically exist at the URL location that you specify in the SAS Deployment Wizard before beginning the SAS Installation. However, knowing the intended value of this URL is important because every client installation is prompted for this value. If you do not specify the URL when SAS Forecast Server is installed, then as a post-installation task, you must manually edit a file on every client machine to specify

this URL. Therefore, it is beneficial to decide on a value for this URL during your planning process so that it can be provided to administrators that might be performing an installation.

For more information about the structure of this file, see "Configuring the SAS Environment File" in the *SAS Intelligence Platform: Middle-Tier Administration Guide*, located at **http://support.sas.com/documentation/onlinedoc/intellplatform**.

Chapter 4

Installation of the SAS Forecast Server

Deployment Overview

Deployment Outline

To install and deploy SAS Forecast Server:

1. If you are upgrading from a previous release of SAS Forecast Server, see Chapter 2, "Migration Tasks," on page 13.
2. Complete Chapter 3, "Pre-Installation Tasks," on page 25.
3. Install the SAS Intelligence Platform, and SAS Forecast Server on each host machine as outlined in your plan.xml file that you developed with your SAS representative.
4. Complete the "Post-Installation Tasks for the Middle Tier" on page 55 for SAS Forecast Server.
5. Complete the tasks to verify the deployment of SAS Forecast Server. For more information, see Chapter 7, "Verify the SAS Forecast Server Installation," on page 59.
6. Complete tasks to secure file system and metadata objects. For more information, see Chapter 9, "Security Administration Tasks," on page 67.
7. Complete tasks to configure reports for use in SAS Forecast Studio. For more information, see Chapter 10, "Report Administration Tasks," on page 75.

Default File Locations

The SAS Deployment Wizard installs and configures your SAS software. The application installation files are installed in a default location referred to as *SAS_HOME*. For example, on a Windows system *SAS_HOME* is `C:\Program Files\SAS`.

The configuration files are stored in a default location referred to as the SAS Configuration directory or *SAS_CONFIG*. For example, on a Windows system *SAS_CONFIG* is **C:\SAS\Config\Lev<*number*>**.

Note: You can deploy up to 10 configurations of the SAS products. The SAS Deployment Wizard specifies each configuration under a **Lev<*number*>** folder. For example, if you deploy a level-2 configuration, then the default configuration directory is **C:\SAS\Config\Lev2**.

The following table shows the default location of the installation and configuration files for SAS Forecast Server.

File	Windows Location	UNIX Location
SAS_HOME	**C:\Program Files\SAS**	**../SAS**
!SASROOT	**C:\Program Files\SAS\SAS Foundation\9.3**	**../SAS/SAS Foundation/9.3**
SAS_CONFIG *Note:* This is the default value. The path might vary by individual installations.	**C:\SAS\Config\Lev<*number*>**	**../SAS/Config/Lev<*number*>**
The SAS Forecast Server batch macros	***!SASROOT*\forecastbat\sasmacro**	***!SASROOT*/forecastbat/sasmacro**
The SAS Forecast Server middle tier	***SAS_HOME*\SASForecastServer MidTier\4.1**	***SAS_HOME*/SASForecastServer MidTier/4.1**
SAS Forecast Studio	***SAS_HOME*\SASForecastStudio\4.1**	Not supported on UNIX

Installation and Configuration of SAS Forecast Server

How to Install and Configure SAS Forecast Server

When you deploy SAS Forecast Server, you also deploy the following components:

- SAS Forecast Studio
- SAS Forecast Server Plug-in for SAS Management Console
- SAS Forecast Server internal macros

Note: The SAS Forecast Server works with the SAS Intelligence Platform, which is also installed and configured when you deploy SAS Forecast Server.

You use the SAS Deployment Wizard to install and configure your software. For more information, see the *SAS Deployment Wizard: User's Guide* at **http://**

support.sas.com/documentation/installcenter/en/ikdeploywizug/62130/PDF/default/user.pdf.

To install and configure SAS Forecast Server:

1. For each machine on which you need to install software, log on to the machine.
   - (Windows) Log on as any user who is in the Administrators group.
   - (UNIX) Log on as a SAS user (for example, sas) that you defined in the SAS Intelligence Platform pre-installation tasks.

   Note: It is recommended that you do not log on as root to perform an installation on a UNIX system.
2. Start the SAS Deployment Wizard from your SAS Software Depot. For example, on a Windows system, double-click the setup.exe file located in your SAS Software Depot folder.
3. In the Choose Language dialog box, select a language from the drop-down list in which to view the SAS Deployment Wizard, and click **OK**.
4. After the SAS Deployment Wizard opens, enter the following information about each screen:
   a. In the Select Deployment Task step, select **Install SAS software**, and then click **Next**.
   b. In the Select Deployment Type step, select **Perform a Planned Deployment**, and then select **Install SAS Software** and **Configure SAS Software.** Click **Next**.

      Note: You can choose to perform these steps individually.
   c. In the Specify Deployment Plan step, select **Specify the full path to a customized deployment plan**. This is the plan.xml file. Typically, the deployment plan is stored in the plan_files folder of the SAS Software Depot. Type the location to this file or click **Browse** to navigate to your plan.xml file. Click **Next**.

      Note: The plan.xml file is generated by the SAS Planning Application. It is included with the pre-installation checklist for your software order. For more information, see the *QuickStart Guide: SAS 9.3 Planned Deployments Using Electronic Software Delivery* at **http://support.sas.com/documentation/installcenter/en/ikqsgplanesd/64415/PDF/default/quickstart.pdf**.
   d. In the Select Deployment Step and Products to Install step, the products that can be installed for the machine that is specified in the **Deployment Step** drop-down box (based on the plan file that you have selected) are listed in the lower pane. Specify which products you want to install. A checkmark to the left of a product indicates that it will be installed. Clear the check box to prevent the product on the same line from being installed. Click **Next**.

      The following products are installed for SAS Forecast Server:
      - SAS Forecast Server Batch Interface for Java
      - SAS Forecast Studio
      - SAS Forecast Server Middle Tier
      - SAS Forecast Server Remote Java Platform Services
      - SAS Forecast Server Plug-ins for SAS Management Console

Note: For a multiple machine deployment, these components could be across several machines.

e. In the Specify SAS Installation Data File step, specify the path to your SID file. By default, the SID file is located in the SAS Software Depot under the sid_files directory. The products that appear in your SAS Software Order e-mail should match the list of **Licensed SAS Foundation Software** on this page. Click **Next**.

f. In the Select Language Support step, select the languages that you would like to install for the products listed. Only those languages that work with your locale are installed. Click **Next**.

g. In the Select Regional Settings step, select a region format from the **Language (Region) [Locale]** drop-down list to specify how your SAS deployment will display text, numbers, currencies, and so on. To configure a Unicode server, select the option to Configure as a Unicode server. Click **Next**.

h. In the Select Java Runtime Environment step, select either to use the recommended Java Run-Time environment, or click **Browse** to navigate to Java Run-Time environment. This page appears if the SAS software that you are preparing to install requires a Java Runtime Environment that the SAS Deployment Wizard cannot find on your machine. This page enables you to determine how you will address the missing Java Runtime Environment. Select to use the recommended Java Runtime Environment, or browse to an environment that is already installed on your machine. Click **Next**.

i. In the Select Microsoft Office Applications step, select the Microsoft Office applications that you want to activate the SAS Add-In for Microsoft Office for. Click **Next**.

j. In the SAS Environments URL, specify a URL location of the SAS environment file, named sas-environment.xml (for example, **`http://<your HTTP server>/sas-environment.xml`**). For more information about how this setting applies to SAS Forecast Server, see "Determine the Location of the SAS Environment URL" on page 28.

k. In the Checking System step, the SAS Deployment Wizard examines your system for required disk space and loads installation files. If the system check reports a problem, follow the directions on this page to remediate it. After the system check is successful, click **Next**.

l. In the Review Required Software step, review the required software for the deployment, and click **Next**.

m. In the Specify Software Location step, specify any required third-party software (for example, a Web application server or Java Development Kit) that is installed so that the installation of your SAS software will be interrupted as infrequently as possible. Enter the location of the named software in the box, or click **Browse**. Click **Next**.

5. Enter the following information needed to configure the software:

a. In the Select Configuration Prompting Level step, select **Typical** to display the basic set of configuration settings. For fewer options, select **Express**. For more granular configuration options, select **Custom**. To configure SAS Forecast Server, the **Typical** setting is recommended. Click **Next**.

b. In the Specify Configuration Information step, create your software's configuration directory. The SAS Deployment Wizard allocates up to 10 configuration levels. The default settings create a level-1 configuration. For example, on a Windows system, this is **`C:\SAS\Config\Lev1`**. If you create a

level-2 configuration, then the default Windows location for it is **`C:\SAS \Config\Lev2`**. Click **Next**.

c. In the Local Machine Name step, type **Fully-qualified Local Host Name** and the **Short Local Host Name**. Click **Next**.

d. Select a migration option. Do one of the following:

- If you are not migrating from the previous release, then do not select **Perform migration**. Click **Next**.
- If you are migrating from the previous release, do not continue with the remaining installation and configuration steps until you review the information in Chapter 2, "Migration Tasks," on page 13.

Note: Information about the configuration settings for the SAS Intelligence Platform and other SAS solutions is beyond the scope of this guide. For more information about these settings, click **Help** in the SAS Deployment Wizard, or see the following:

- For information about entering SAS Deployment Wizard information for the SAS Intelligence Platform, see **`http://support.sas.com/documentation/onlinedoc/intellplatform`**.
- For information about entering SAS Deployment Wizard information for other solutions, see the solution documentation.

6. Enter the information needed to configure SAS Forecast Server:

Note: The Environment Setup options in steps b through d are not available when performing a migration. The environments are created based on the migrated content.

a. In the SAS Forecast Server: SOAP Configuration step, select **Grant access to SAS Forecast Studio tasks** to allow access to these tasks by SAS Add-In for Microsoft Office and SAS Enterprise Guide. Click **Next**.

b. In the SAS Forecast Server: Environment Setup step, you can select to **Create an environment during configuration**. This step lets you avoid manually creating an initial environment in SAS Management Console before using SAS Forecast Studio. An environment is required as a container for SAS Forecast Server projects. If you choose to skip this step, you must manually create an environment using the SAS Forecast Server Plug-in for SAS Management Console before using SAS Forecast Studio. Click **Next**.

c. In the **SAS Forecast Server: Environment Dependencies** step, select a configured instance of the SAS Workspace Server that will host the environment. The SAS Forecast Server depends on a configured instance of the SAS Workspace Server to successfully complete its configuration of the environment. Click **Next**.

Note: Do not use SASMeta as the workspace server in this step. For example, select SASApp instead of SASMeta.

Note: If there is a single workspace server configured in the system, then that server will be assumed as the host for the environment and this window does not appear.

d. If you chose to create an environment, the **SAS Forecast Server: Environment Setup** screen opens. Complete the following:

- Type a name for the environment in the **Forecast Environment Name** field, or use the default name provided.

- Type a description for the environment in the **Environment Description** field, or use the default description provided.
- Type the file system location in the **Content Location** field to specify the root path of the SAS Forecast Server project directory (also referred to as the content location), or use the default location provided. Even if you accept the default path, you must manually create the project directory on the workspace server. The SAS Forecast Studio users must have full access to this location. For more information about specifying permissions for environments, see "Security Administration Tasks" on page 67

  Note: This field is populated with a default value that configures the environment's path under the SAS configuration directory. For example, on a Windows system, the default content location is **`C:\SAS\Config\Lev1\AppData\SASForecastServer4.1`**. If you intend to keep your projects in a different location, you should change this path to a location outside of the configuration directory. Specifying an outside location keeps the project data separate from the configuration data and will also make future migrations easier.

e. Click **Next**.

f. In the SAS Internal Account: Forecast Server Metadata User step, specify the internal password that is used for the fsmeta internal user. This internal account is used by SAS Forecast Server to read and update metadata. Click **Next**.

7. In the Deployment Summary, review the list of products that you are about to install, and click **Start**. The SAS Deployment Wizard switches from its information gathering mode to its installation and configuration mode. It performs the following tasks:

   - Installation of your software. After you complete the System Requirements Wizard, the SAS Deployment Wizard does not require any user interaction. The SAS Deployment Wizard installs each product that is shown in the list in the order shown. The installations are chained, so you do not need to initiate the installation programs. After installing the first product, the SAS Deployment Wizard automatically proceeds with the installation of the next product. On a Windows system, the default location for the installation files is in the **`C:\Program Files\SAS`** folder.
   - Configuration of your software. The SAS Deployment Wizard attempts to configure all the software that you selected to install.

8. The SAS Deployment Wizard indicates that the installation and configuration of your SAS software is complete in the **Deployment Complete** screen. Each software component should have a check mark next to it in the list. Click **Next**.

9. In the Additional Resources step, review the manual configuration instructions in the Instructions.html file. This file is saved to your SAS configuration directory during deployment (for example, **`C:\SAS\Config\Lev1\Documents\Instructions.html`**). You can also view the list of links for additional resources about your deployment. As an option, you can print the list of links for later reference. Click **Finish** to exit the SAS Deployment Wizard.

10. Complete the post-installation tasks and the administration tasks for SAS Forecast Server.

Chapter 5

Post-Installation Tasks for the Server Tier

Creating Operating System Accounts in UNIX Environments

Using Operating System Groups to Assign Permissions

Users have different operating system privileges on the SAS Workspace Server. By defining a user group for SAS Forecast Server, you can assign all SAS Forecast Users to the same group and grant the same permissions to all SAS Forecast Server users at one time. All SAS Forecast Server users must have Read, Write, and Execute permissions for each environment directory that a user is permitted to use. Users also need permissions to all of the files and directories in an environment directory. The operating system must be configured to grant these permissions as new files and directories are

created.. The exact details of how to do this depends on which operating system groups are defined and your site's security policies.

Conditions for the User Group for SAS Forecast Server

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

- A group of forecasting users is created for the UNIX operating environment. The logon IDs for each SAS Forecast Server user must be in this group. The group must also include any user who might run code that is created from a SAS Forecast Server project in a SAS session.
- Users can be members of multiple groups, but the SAS Forecast Server group is the primary group for each user.
- The SAS scripts are updated to grant permissions to the SAS Forecast Server users on the SAS Workspace Server and SAS Stored Process Server. For more information, see "Update the SAS Scripts to Grant Permissions to the User Group" on page 38.
- Each environment directory has the correct ownership, and the user group for SAS Forecast Server has Read, Write, and Execute permissions.

Update the SAS Scripts to Grant Permissions to the User Group

Using the **umask** option, you can grant permissions to the SAS Forecast Server users on a conditional basis if the user is part of the SAS Forecast Server user group.

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

To set these permissions:

1. On each SAS Workspace Server, open **<*config-dir*>/Lev1/SASApp/appservercontext_env.sh**.
2. Enter the configuration information for your operating environment. Here is the general format of this code:

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

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

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

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

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

   Here are code examples for each UNIX environment where SAS Forecast Server is supported:

Operating Environment	Sample Code

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

H64 (HP-Risc)	`CMD=/usr/bin/id`
	`CURR_GID=‘eval $CMD -g‘`
	`GID=201 if [ $CURR_GID -eq $GID ] ; then umask 002`
	`fi`

H64I (HP-Itanium)	`CMD=/usr/bin/id`
	`CURR_GID=‘eval $CMD -g‘`
	`GID=201 if [ $CURR_GID -eq $GID ] ; then umask 002`
	`fi`

S64 (Solaris)	`CMD=/usr/xpg4/bin/id`
	`CURR_GID=‘eval $CMD -g‘`
	`GID=201 if [ $CURR_GID -eq $GID ] ; then umask 002`
	`fi`

SAX (Solaris for X64)	`CMD=/usr/xpg4/bin/id`
	`CURR_GID=‘eval $CMD -g‘`
	`GID=201`
	`if [ $CURR_GID -eq $GID ] ; then umask 002`
	`fi`

LNX (Linux)	`#!/bin/bash`
	`CMD=/usr/bin/id`
	`CURR_GID=‘eval $CMD -g‘`
	`GID=500`
	`if [ "$CURR_GID" -eq "$GID" ] ; then umask 002`
	`fi`

Managing Roles and Capabilities

About Metadata Groups, Roles, and Capabilities

Default Roles for SAS Forecast Server

SAS Forecast Server ships with default metadata groups and roles. When SAS Forecast Server is deployed, capabilities are already assigned to the default roles. Using SAS Management Console, you can add additional capabilities to these default roles. You can also create your own metadata groups and roles. Then you specify the capabilities for that role.

These roles are available for an out-of-the-box deployment of SAS Forecast Server:

Roles	Description
Forecast Server: Administrator	The capabilities that enable users to manage product content, such as projects, are assigned to this role.
Forecast Server: Forecaster	The capabilities that enable users to create projects and generate forecasts are assigned to this role.
Forecast Server: Analyst	The capabilities that enable the user to analyze the time series data and the results of the forecast are assigned to this role.
Forecast Server: Browser	The capabilities that enable the user to view the results of the forecasts and use reports are assigned to this role.

In addition to these SAS Forecast Server roles, you might also use the Job Execution: Job Submitter and Management Console: Advanced roles. These roles are available in SAS Management Console and are not specific to SAS Forecast Server. The Job Execution: Job Submitter role enables you to run stored processes from the Reports and Stored Processes dialog box. The Management Console: Advanced role enables you to access the product plug-ins (such as the SAS Forecast Server plug-in) in SAS Management Console.

Capabilities for SAS Forecast Server

The following table lists the capabilities for SAS Forecast Server. Use capabilities to restrict the ability of a user to request that SAS Forecast Server perform an action, such as generate forecasts or update a model specification.

Table 5.1 *Client Access*

Capability	Description
Forecast Studio	Enables the use of the SAS Forecast Studio client.
Plug-in for Management Console	Enables the use of the SAS Forecast Server Plug-in for SAS Management Console.
SOAP Service Bridge	Enables the use of the SAS Forecast Studio tasks that are available in SAS Enterprise Guide and the SAS Add-In for Microsoft Office.
Macro Bridge	Enables the use of the SAS Forecast Server macros (for example, FSCOPY).

Table 5.2 *General Features*

Capability	Description
Administer Product	Enables you to administer the metadata for SAS Forecast Server.
Analyze Time Series	Enables the Series View in SAS Forecast Studio.
Analyze Models	Enables the use of analysis tests in the Modeling View in SAS Forecast Studio.
Change Series Usages	Enables the user to specify whether a series is active. For example, this capability enables the **Active Series** check box in the Forecasting View and the Modeling View.
Import New Data	Enables a user to choose whether SAS Forecast Server uses the most up-to-date data when generating forecasts. For example, this capability enables the **Use updated data if available** check box in several dialog boxes in SAS Forecast Studio, such as the Update Project Version dialog box and the Reforecast Project dialog box.

Table 5.3 *Forecasts*

Capability	Description
Modify Forecasts	Enables a user to forecast the project or series in SAS Forecast Studio.
Reconcile Forecasts	Enables a user to specify whether SAS Forecast Server should try to reconcile the hierarchy. For example, this capability enables the **Reconcile Hierarchy** menu item, the Reconcile icon in the Forecasting View, and the **Reconcile** message (that appears in the workspace if there are override conflicts or unresolved nodes).
Override Forecasts	Enables a user to create, edit, or delete override values. For example, this capability enables the Overrides Calculator, the ability to lock overrides, and the ability to set scenario forecast values as Overrides in the Scenario Analysis View.

Table 5.4 *Models*

Capability	Description
Change Model Selection	Enables a user to change the baseline model for the series. For example, in the Modeling View, this capability enables the **Set this model as forecast model** and **Reset to "automatic selection"** links.
Modify Models	Enables a user to create, delete, copy, and edit models. This capability also enables the user to import models from a catalog and export models to a catalog.

Table 5.5 *Events*

Capability	Description
Change Event Usage	Enables the user to specify if an event is used in the model. For example, this capability enables the **Usage in system-generated models** drop-down list in the Events Properties dialog box.
Modify Events	Enables a user to create, delete, copy, and edit events. This capability also enables the user to import events from a catalog and export events to a catalog.

Table 5.6 *Environments*

Capability	Description
View Environment Details	Enables a user to view the properties of SAS Forecast Server environments.
Manage Environment Settings	Enables a user to edit the properties of SAS Forecast Server environments.
Manage Environments	Enables a user to create, delete, rename, register, and unregister environments.

Table 5.7 *Projects*

Capability	Description
Create Projects	Enables a user to create a new project. This capability enables the New Project Wizard.

Capability	Description
Manage Projects	Enables a user to delete, copy, register, and unregister a project; import and unarchive a project; export and archive a project; delete project archives; and update the project version.
Manage Project Access	Enables a user to specify whether a project is shared and to change the ownership of a project. For example, this capability enables the **Change Owner** and **Public access** options in the SAS Forecast Server Plug-in for SAS Management Console.
Manage Project Settings	Enables a user to specify the hierarchy and variable settings and forecasting settings for a project. For example, this capability enables the Hierarchy and Variable Settings and Forecasting Settings dialog boxes in SAS Forecast Studio.
Modify Project Scripts	Enables a user to edit the project code that runs when the project is opened and when the project is closed. For example, this capability enables the user to edit the code in the Start-up and Shutdown Code dialog box.
View Project Scripts	Enables a user to view the project code that was written to run when the project is opened and when the project is closed. For example, this capability enables the **Start-up and Shutdown Code** button in the New Project Wizard and in the Project Properties dialog box.

Table 5.8 *Notes*

Capability	Description
View Notes	Enables the Notes panel at the bottom of the Forecasting View.
Modify Notes	Enables a user to edit the content in the Notes panel at the bottom of the Forecasting View. This capability also enables the user to create and remove notes.

Table 5.9 *Reports*

Capability	Description
Use Reports	Enables the **Reports and Stored Processes** menu item and the Reports and Stored Processes dialog box in SAS Forecast Studio.

Capability	Description
Manage Reports	Enables a user to manage reports by using the SAS Forecast Server Plug-in for SAS Management Console. For example, this capability enables a user to deploy and remove sample reports.

Table 5.10 *Scenarios*

Capability	Description
View Scenarios	Enables the Scenario Analysis View in SAS Forecast Studio.
Modify Scenarios	Enables a user to create new scenarios and edit, save, or delete existing scenarios.

Table 5.11 *Special Features*

Capability	Description
Legacy Features	Enables legacy features. This option is provided only to aid users through transitional periods that result from design changes in the next product release. Support for these legacy features is limited. You should enable these features only at the direction of SAS Technical Support.
Experimental Features	Enables experimental features that are still under development and might change (or be removed) in a future release. Also, no migration support will be provided for these features. You should enable these features only at the direction of SAS Technical Support.
Debugging Features	Enables debugging features that are provided to assist customers and SAS Technical Support when problems arise at your site. You should enable these features only at the direction of SAS Technical Support.

How to Configure the SAS Forecast Server Users

Note: On each SAS Workspace Server that contains forecasting environments, you must have an operating system account for each user of SAS Forecast Server.

You can use the default roles that ship with SAS Forecast Server to quickly configure your SAS Forecast Server users. Each default role is assigned capabilities that allow access to various software features.

The following steps configure an administrator account for SAS Forecast Server by using the default roles. In this example, all SAS Forecast Server roles are listed. All access was the typical configuration for administrator accounts in releases prior to 4.1.

Starting in SAS Forecast Server 4.1, the defining quality of an administrator account is the Administer Product capability that grants the administrator account expanded privileges under the product security model. The Administer Product capability is granted by the Forecast Server: Administrator role.

To configure the metadata accounts:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
2. Create a metadata account for the user in SAS Management Console and associate that account with the operating system account.
3. In the User Manager, right-click the user and then select **Properties**. The Properties dialog box appears.
   a. Select the **Groups and Roles** tab.
   b. Assign the user to a default role or a role that you created for your site.

      The Forecast Server: Administrator role is sufficient if the administrator account only needs to manage actions in SAS Forecast Server. However, this role does not grant access to all product features. To give the administrator access to all product features, you must select all of the following roles:

      - Management Console: Advanced
      - Job Execution: Submitter
      - Forecast Server: Administrator
      - Forecast Server: Analyst
      - Forecast Server: Browser
      - Forecast Server: Forecaster

      TIP To view the capabilities for a role, right–click the role and select **Properties** from the pop-up menu.

   c. Click  to move the item to the **Member of** list.
   d. Click **OK**.

Creating and Configuring a SAS Forecast Server Environment

About SAS Forecast Server Environments

Note: A SAS Forecast Server environment is different from a SAS environment. A SAS environment is used only when you log on. The SAS Forecast Server environment is used after you log on to the product and start a product session.

A SAS Forecast Server environment is a product workspace for product sessions. These environments are created by the SAS Forecast Server administrator and are used only by

SAS Forecast Server and its client applications. Environments can be used to organize your projects and to control access to SAS Forecast Server projects.

If you have more than one environment, do not use the same project directory for multiple environments. In general, a file system location should be used by a single environment. This restriction is necessary to ensure the integrity of the file system content and to properly coordinate client access. In addition, do not configure one environment to use a subdirectory within a different environment.

CAUTION:

You must create at least one environment before you can create a project in SAS Forecast Studio. If you opted to create the default environment in the SAS Deployment Wizard, you do not need to create another environment in SAS Management Console. However, you must create a project directory for the environment and set the appropriate permissions for that project directory.

Create a SAS Forecast Server Environment in SAS Management Console

To create an environment:

1. Open SAS Management Console and log on as the product administrator for SAS Forecast Server.
2. In the navigation tree, locate the **Application Management** node.
3. Right-click the **Forecast Server** node, and select **New Environment**. The Create Environment dialog box appears.
4. Specify a name for the new environment.
5. Select a workspace server. By default, the first available workspace server is selected.

   Note: Do not select SASMeta as the workspace server.
6. In the **Location** field, specify the root path to the project directory.

   Note: The project directory should be used only by this environment.
7. Specify the location of the **Reports** folder in the metadata. This path must already exist. If you specify a blank value, then the support for reports is disabled for that environment.

   Note: The **Reports** folder requires full security permissions for the Forecast Server Metadata User.
8. Click **OK**.

Set File System Permissions on Windows

In Windows operating environments, you must set file system permissions for all of the SAS Forecast Server users. To support the public settings on projects, users must have full control of and access to the environment directory and its subdirectories.

To set file system permissions on Windows for all users:

1. Open Windows Explorer and select the root directory for the SAS Forecast Server environment.
2. Right-click on this directory, and then select **Properties**.

3. Select the **Security** tab.
4. Select the users for the current machine.
5. Enable **Full Control** for the specified group of users.
6. Click **OK**.

TIP If you plan to use environments only for the convenience of organizing projects, you can create each environment directory under a common parent directory that is configured with the appropriate file system permissions. In this way, the child directories inherit the permissions from the parent directory, so you do not need to configure each environment.

Creating and Configuring Libraries

Understanding Libraries

Overview of Libraries

SAS Forecast Server uses SAS libraries and data sets to manage and access project data. For more information about libraries, see the "SAS Libraries" chapter in *SAS Language Reference: Concepts* and the "LIBNAME Statement" topic in *SAS Statements: Reference*.

Note: Library names cannot begin with an underscore character. This naming convention is reserved for internal SAS libraries.

In SAS Forecast Server, libraries are either assigned by the SAS Forecast Server application or external to the SAS Forecast Server application. Library assignments that are made by the system or a user are examples of external library assignments.

External Library Assignments

Libraries can be assigned when a server session is created or in the start-up code for a specific environment. Because these library assignments are made outside of SAS Forecast Server, they are considered external library assignments.

Here are four main sources of external library assignments:

- All SAS libraries that ship with SAS are automatically assigned to each SAS session. Examples of Base SAS libraries are Sashelp, Sasuser, and Work.
- Additional libraries can be assigned by adding LIBNAME statements to your SAS or server configuration files. The scope of the configuration file (in other words, is it a SAS or server configuration file) determines the availability of the library.
- Libraries can also be assigned by including LIBNAME statements in the start-up code for an environment. These libraries are available only for the associated environment. If you assign a library in your environment's start-up code, you must include a corresponding LIBNAME CLEAR statement in the shutdown code. For more information, see "Adding Start-Up and Shutdown Code for an Environment" on page 103.
- Pre-assigned libraries that are defined in the metadata and associated with a given server are automatically assigned when the server session is created.

Libraries Assigned by SAS Forecast Server

SAS Forecast Server enables you to manage and assign libraries on-demand. Even if these libraries are not assigned, they appear in SAS Forecast Studio like any other library. When SAS Forecast Server detects that a library is going to be accessed (for example, when the library is referenced in the code), SAS Forecast Server assigns the library before allowing that access to occur.

Due to security considerations, these libraries are not available in SAS Forecast Studio by default. SAS Forecast Server includes a configuration setting to enable the use of each type of on-demand library: manually assigned metadata libraries, configured environment libraries, and automatic environment libraries. For more information, see "Enable Expanded Support for Libraries" on page 49.

Here are the three types of libraries that are assigned by SAS Forecast Server:

- Libraries that are defined in the metadata but are not pre-assigned. When SAS Forecast Server detects that one of these libraries needs to be used, the library is assigned using the META engine with the METAOUT=DATA option.

  Note: The META engine requires that each data set (also referred to as a table) is registered in the metadata. If the data set is not registered, then the data set does not appear in SAS Forecast Studio. Data sets that exist only as files on the file system cannot be accessed from SAS Forecast Studio when these data sets are located in a metadata defined library that is not pre-assigned.

- Configured environment libraries are defined in the ***environment-directory*****\Config\libs** data set. Each row of this data set represents a library assignment and contains the information required for the BASE engine:

  - the LIBNAME
  - the file system path

    Note: Specify the absolute path to the library directory to prevent SAS Forecast Server from defaulting to the working directory.

  - whether access to the library should be read-only

  When SAS Forecast Server detects that a configured environment library needs to be used, the library is assigned using the BASE engine.

- Automatic environment libraries are implicitly defined by creating a subdirectory for each library in the ***environment-directory*****\Libraries** directory. The name of this subdirectory implies the LIBNAME for the library assignment and the access settings for the library.

  When SAS Forecast Server detects that an automatic environment library needs to be used, the library is assigned using the BASE engine.

The configured and automatic environment libraries enable you perform these tasks:

- manage libraries for users who have access to the file system
- define local libraries for individual environments

Authorization Checks

One key detail of library management and use is the ability to perform authorization checks on the data accesses that are being attempted by the user. How these authorization checks are performed depends on the type of library that the user is trying to access.

Two strategies are used.

- The SAS Intelligence Platform and the SAS Metadata Server require that a user have ReadMetadata permission on a library object in order for the library to be available in SAS Forecast Studio. For libraries that are defined in the metadata, SAS Forecast Server requires Read metadata permission for the user to have Read access and Write metadata permission for the user to have Write access. Permissions can also be set for elements within the library, such as a data set (also referred to as a table) or a data set variable (also referred to as a column). For example, if SAS Forecast Studio tries to read a variable in a data set, then SAS Forecast Server searches for an associated Column, Table, or Library object in that order. The authorization check is performed against the first object that is found.

  TIP You can create table objects for any type of metadata library by using the SAS Forecast Server plug-in for SAS Management Console. For more information, see the Help for the SAS Forecast Server plug-in.

- All other libraries are assigned using the BASE engine (for example, the configured and automatic environment libraries) or are assigned in a way that cannot be determined by SAS Forecast Studio (for example, when a LIBNAME statement is included in a SAS or server configuration file). For these libraries, SAS Forecast Server checks the dictionary tables for the SAS session. Permissions for a library are specified by the Read-Only access setting in the dictionary table.

Enable Expanded Support for Libraries

By default, SAS Forecast Server restricts the use of metadata libraries that are manually assigned, automatic environment libraries, and configured environment libraries. You must configure SAS Forecast Server so that users can access these libraries.

To enable this library support:

1. Open SAS Management Console as the SAS Administrator (for example, sasadm), and then connect to a metadata repository.
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, click **Forecast Server**.
6. From the **Manual-assign metadata libraries**, **Automatic environment libraries**, and **Configured environment libraries** drop-down lists, specify whether SAS Forecast Server allows access to each type of library.

   Click **OK**.

How to Define a Pre-assigned Library in the Metadata

To create a metadata library that is pre-assigned:

1. Create a metadata definition for the new library.
   a. Open SAS Management Console as the SAS Administrator (for example, sasadm), and then connect to a metadata repository.
   b. Expand the **Data Library Manager** node, and then select **Libraries**.

c. Right-click the library that you want to pre-assign, and then select **Properties**.

d. Select the **Options** tab.

e. Click **Advanced Options**.

f. Select the **Library is preassigned** check box. The selected library is assigned whenever a session is created for one of the assigned servers.

g. Ensure that the library is assigned to the correct SAS servers.

h. Click **OK**.

2. Set the metadata permissions on the new library object for your SAS Forecast Server product administrator and users.

   a. In SAS Management Console, expand the **Data Library Manager** node, and select **Libraries**.

   b. Right-click the library name, and then select **Properties**.

   c. Select the **Authorization** tab.

   d. In the **Users and Groups** area, select the product administrator and grant permissions by selecting the **Grant** boxes. Repeat this step for the SAS Forecast Server users.

      Verify that the fsmeta account has ReadMetadata permission on the library. If the fsmeta account does not have this permission, SAS Forecast Server might not detect the library or might classify the library incorrectly.

3. If the SAS Object Spawner is running, stop and restart the SAS Object Spawner.

How to Define a Metadata Library That Is Assigned by SAS Forecast Server

To create a metadata library that is not pre-assigned:

1. Verify that SAS Forecast Server is configured to support automatic environment libraries. For more information, see "Enable Expanded Support for Libraries" on page 49.

2. Create a metadata definition for the new library:

   a. Open SAS Management Console as the SAS Administrator (for example, sasadm@saspw), and then connect to a metadata repository.

   b. Expand the **Data Library Manager** node, and then select **Libraries**.

   c. Right-click the library that you want to pre-assign, and then select **Properties**.

   d. Select the **Options** tab.

   e. Click **Advanced Options**.

   f. Ensure that the library is assigned to the correct SAS servers.

   g. Click **OK**.

3. Set the metadata permissions on the new library object for your SAS Forecast Server product administrator and users.

   a. In SAS Management Console, expand the **Data Library Manager** node, and select **Libraries**.

   b. Right-click the library name, and then select **Properties**.

c. Select the **Authorization** tab.

d. In the **Users and Groups** area, select the product administrator and grant permissions by selecting the **Grant** boxes. Repeat this step for the SAS Forecast Server users.

   Verify that the fsmeta account has ReadMetadata permission on the library. If the fsmeta account does not have this permission, SAS Forecast Server might not detect the library or might classify the library incorrectly.

4. Register the data sets that you want to include in this library.

Create an Automatic Environment Library

To create an automatic environment library:

1. Verify that SAS Forecast Server is configured to support automatic environment libraries. For more information, see "Enable Expanded Support for Libraries" on page 49.
2. In the ***`environment-directory`*`\Libraries`** directory, create a new directory for the automatic environment library. The library name should be the same as the desired libref. For example, if the library name is Hpfuser, then the name of the directory should be **`hpfuser`**.

   Note: If the **`Libraries`** directory does not exist, then you need to create it.
3. Specify the security access for the library. By default, the libraries are assigned read-write access. If the library should have Read-Only access, append **`.r`** to the directory name. For example, the hpfuser.r directory creates the Hpfuser library with Read-Only access.

Now, when a user opens this environment in SAS Forecast Studio, the new library should be available.

Create a Configured Environment Library

To create a configured environment library:

1. Verify that SAS Forecast Server is configured to support configured environment libraries. For more information, see "Enable Expanded Support for Libraries" on page 49.
2. In the ***`environment-directory`*`\Config`** directory, create a libs.sas7bdat file. This data set should contain three character variables: LIBNAME, Path, and Readonly.

   Note: If the Libs data set does not exist, SAS Forecast Server tries to create it when a user opens this environment in SAS Forecast Studio.
3. Edit the Libs data set to include the information for the new library. For example, to define the Hpfuser library, you might use these values:
   - For the LIBNAME variable, specify **`hpfuser`**.
   - For the Path variable, specify **`C:\mylibs\hpfuser`**.
   - For the Readonly variable, specify **`yes`**.

Now, when a user opens this environment in SAS Forecast Studio, the new library should be available.

Configuring the SAS Forecast Studio Tasks

About the SAS Forecast Studio Tasks

The Forecast Studio Create Project task, the Forecast Studio Open Project task, and the Forecast Studio Overrides Submit task enable you to create and work with SAS Forecast Studio projects in SAS Enterprise Guide and the SAS Add-In for Microsoft Office. The SAS Add-In for Microsoft Office extends the functionality of Microsoft Excel, Microsoft Word, and Microsoft PowerPoint by enabling you to access SAS analytics and SAS reporting functionality without any SAS programming experience. The SAS add-in is designed for people who are familiar with these Microsoft Office programs but who might be new to SAS.

Prerequisites for Using the SAS Forecast Studio Tasks

To use the SAS Forecast Studio tasks in the SAS add-in, you must complete these tasks:

- install SAS Forecast Server 4.1.
- install SAS Enterprise Guide 4.3 or the SAS Add-In 4.3 for Microsoft Office on each client machine, which is a product in the SAS Enterprise Business Intelligence Server bundle.

  Note: You must apply hot fixes C44006 and C46006 to SAS Enterprise Guide 4.3 and the SAS Add-In 4.3 for Microsoft Office in order for the SAS Forecast Studio tasks to work.
- configure SAS Forecast Server to use the SAS Add-In for Microsoft Office.

TIP To avoid having to reconfigure the product, make sure that you enable the use of the SOAP service bridge for SAS Forecast Server during deployment.

Grant Access to the SAS Forecast Studio Tasks

Note: When you initially configured SAS Forecast Server, you might have selected the **Grant access to SAS Forecast Studio tasks** in the SAS Forecast Server: SOAP Configuration step of the SAS Deployment Wizard. This option is required to enable the use of these tasks in SAS Enterprise Guide and the SAS Add-In for Microsoft Office. If you selected this option, then skip the following reconfiguration steps.

If you did not grant access to the SAS Forecast Studio tasks when you initially configured SAS Forecast Server, you must reconfigure SAS Forecast Server before you can use these tasks in SAS Enterprise Guide or the SAS Add-In for Microsoft Office. You do not need to unconfigure SAS Forecast Server first. You simply need to complete the following steps to reconfigure the product.

To reconfigure SAS Forecast Server to grant access to the SAS Forecast Studio tasks:

1. To reconfigure SAS Forecast Server, open the SAS Deployment Wizard from your SAS Software Depot. For example, on a Windows system, double-click the setup.exe file located in the SAS Software Depot folder.
2. In the SAS Deployment Wizard, complete these steps:

a. In the Select Deployment Task step, select **Install SAS software**.

b. In the Select Deployment Type step, select **Perform a Planned Deployment**, and then select **Configure SAS Software**.

c. In the Select Configuration Prompting Level step, select **Typical** to display the basic set of configuration settings.

d. In the Select Products to Configure step, click **Clear All**, and then select **SAS Forecast Server Mid-Tier**. You must also reconfigure your Web application server, so you must stop your Web application server when reconfiguring the product.

e. In the SAS Forecast Server: SOAP Configuration step, select **Grant access to SAS Forecast Studio tasks**.

f. Navigate through the remaining windows in the SAS Deployment Wizard, and click **Finish**.

Chapter 6

Post-Installation Tasks for the Middle Tier

Changing Configuration Properties in SAS Management Console

Several configuration properties exist for the SAS Forecast Server middle tier. You can set these properties by using the Configuration Manager in SAS Management Console. The values of these configuration properties are cached at run time, so if you change the value of a configuration property, you must restart SAS Forecast Server. For more information about the Configuration Manager, see the SAS Management Console Help.

Note: If your site allows JMX access, then you can use the JMX calls to indicate when the application should reload the configuration properties. These signals eliminate the need for restarting SAS Forecast Server for new configuration properties to take effect. For more information, see "JMX Beans" on page 166.

Specifying a Time Out

Specify a Time Out for SAS Forecast Studio

By default, a SAS Forecast Studio session times out after being idle for 60 minutes. The SAS Forecast Server checks every 5 minutes to see whether the session has timed out. You can change the value of the time out and how frequently SAS Forecast Server checks a session.

To specify a new time-out value for an idle SAS Forecast Studio session:

1. Start SAS Management Console and connect as a SAS administrator.
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, click **SAS Forecast Server**.
6. Specify new values for the **Idle timeout limit** and **Time-out check interval** options.

   Click **OK**.
7. Restart SAS Forecast Server for these changes to take effect.

Specify a Time Out for the SOAP Bridge

The SOAP bridge is used to run the SAS Forecast Studio tasks in SAS Enterprise Guide and the SAS Add-In for Microsoft Office. By default, this connection times out after being idle for 60 minutes. The SAS Forecast Server checks every 60 seconds to see whether the service has timed out. You can override the time-out value by specifying the **`com.sas.analytics.forecasting.soap.timeout`** Java system property. If you set this property to a value less than or equal to 0, then the time-out is disabled, and you must restart the SOAP service to delete any old sessions.

Configuring and Using Java Web Start

Customizing the Default URL

By default, you can launch SAS Forecast Studio from **`http://`*`your-server-name`*`:`*`your-http-port`*`/SASForecastServer/main.jnlp`**.

To open a particular project in SAS Forecast Studio, add these parameters to your URL:

- **`forecasting.launch.environment`** — specifies the forecasting environment.

  TIP You are not required to specify the project parameter. You might choose to specify on the environment parameter.
- **`forecasting.launch.project`** — specifies the name of the project.

For example, if your URL is **`http://localhost:8080/SASForecastServer/main.jnlp?forecasting.launch.environment=Default&forecasting.launch.project=Project1`**, then SAS Forecast Studio opens and displays Project 1 in the default environment.

How to Specify the Value of the SAS Environment URL

To specify the link to the Java Web Start client as the SAS environment URL:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).

2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, select **Forecast Studio (Java Web Start)**.
6. In the **SAS environment URL** box, specify the URL for the sas-environment.xml file.

   Click **OK**.
7. Restart the Web application server for these changes to take effect.

Set the Preferred Environment in the Log On Dialog Box for SAS Forecast Studio

When users log on to SAS Forecast Studio, they must specify a SAS environment to use. You can restrict the environments that users have access to and specify a preferred environment that is selected by default.

Note: If the preferred environment does not exist at run time, SAS Forecast Studio assumes that you did not specify a preferred environment.

To specify the list of environments that should appear in the Log On dialog box:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, select **Forecast Studio (Java Web Start)**.
6. In the **Preferred SAS environment** box, specify the name of the SAS environment that should be selected in the Log On dialog box.

   Note: This name is case sensitive.

   Click **OK**.
7. If the user should not be able to change the environment in the Log On dialog box, set the **Lock SAS environment selection** property to `true`.
8. Restart SAS Forecast Server for these changes to take effect.

Configuring the Archive Functionality

Note: With the exception of the compression level, do not modify the archiving properties unless directed to by SAS Technical Support.

The archiving properties in SAS Forecast Server enable you to control the process for creating and extracting project archives.

To configure the archiving properties:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, select **Forecast Server**.
6. Set these archiving properties:

Property	Description	Default Value
Compression level	Specifies how much to compress a new archive.	Default of the system level
Files to ignore (by extension)	Specifies the file types that you do not want to include in an archive.	sas7bndx
Data set files to CPORT (by extension)	Specifies data files that you want to include in the archive. A comma-separated list of values is expected.	sas7bdat
Catalog files to CPORT (by extension)	Specifies the catalog files that you want to include in the archive. A comma-separated list of values is expected.	sas7bcat
Additional CPORT option	Enables you to include additional options in the PROC CPORT statement that is used to create an archive.	(blank)

   Click **OK**.
7. Restart SAS Forecast Server for these changes to take effect.

Chapter 7

Verify the SAS Forecast Server Installation

Start the SAS Servers and the SAS Object Spawner

Before you can verify your installation, you must start the SAS Metadata Server, the SAS Object Spawner, the application server for the Web Infrastructure Platform, and the application server for SAS Forecast Server. For more information, see "Requirements for Starting SAS Forecast Studio" on page 61.

Verify the SAS Forecast Server Installation

To verify your installation:

1. Start SAS Forecast Studio on the Windows machine where you installed the client. From the **Start** menu, select **Programs** ⇨ **SAS** ⇨ **SAS Forecast Studio** ⇨ **SAS Forecast Studio 4.1**.
2. In the Log On dialog box, select the SAS environment and type your user name and password. Click **Log On**.
3. To create a new project when you open SAS Forecast Studio, click **New** in the Projects dialog box. If you closed the Projects dialog box, select **File** ⇨ **New Project**.
4. Specify a name for the project, such as `Test_Project`, and then click **Next**.
5. In the New Project wizard, select a library that contains a data set that you want to forecast. For example, expand the **SASHELP** library.
6. Select the **PRICEDATA** data set, and then click **Next**.
7. Select the classification (BY) variables. For example, select **PRODUCT** and **PRODUCTNAME**, and then click  to move the variables to the **Classification variables selected** list. Click **Next**.

8. Specify the **Time ID** variable and its properties. For example, select **DATE** for the **Time ID** variable. Keep the remaining default settings, and then click **Next**.
9. Assign the **Dependent** role to the **SALE** variable. Select the **SALE** variable, and then select **Dependent** in the **Role** column. Click **Next**.
10. Keep the default values specified to prepare the data for each forecast, and then click **Next**.
11. Keep the default values specified for the optional project settings, and then click **Next**.
12. Click **Finish** to produce the forecast.

After you create the project and forecast the series, your installation is complete and verified.

Chapter 8

Start SAS Forecast Studio

Requirements for Starting SAS Forecast Studio

Before you can run the SAS Forecast Studio client, you must start the servers (for example, the SAS Metadata Server, your Web Application Server, and SAS Remote Services). For information about how to start these servers, see *SAS Intelligence Platform: System Administration Guide*.

Using the Desktop Application

SAS Forecast Studio runs only on the Windows operating system. To start SAS Forecast Studio on the system where it is installed, select **Start** ⇨ **Programs** ⇨ **SAS** ⇨ **SAS Forecast Studio** ⇨ **SAS Forecast Studio 4.1**.

To open SAS Forecast Studio in a particular environment or for a specific project, type **`launchFile=filename`** at the command prompt where *filename* is the path to a .fs file that contains the parameters for your project. An example of a filename is **`C:\Documents and Settings\Europe\Sales\Products2011.fs`**. The **`Products2011.fs`** file contains the following code:

```
environment=Default
project=Products2011
```

When you run the **`launchFile=Documents and Settings\Europe\Sales\Products2011.fs`**, SAS Forecast Studio opens and displays the Products2011 project.

Using Java Web Start

How to Launch SAS Forecast Studio

You do not need to install SAS Forecast Studio on every system where you need to run SAS Forecast Studio. Instead, SAS Forecast Server supports automatic downloads of SAS Forecast Studio by using Java Web Start.

To launch SAS Forecast Studio using the Java Web Start client, perform either of these steps:

- Open the **SAS Forecast Server Status** Web page. The default URL for this page is where the SAS Forecast Server services are deployed. An example of this URL is **`http://`*`your-server-name`*`:`*`port-number`*`/SASForecastServer/Status`**.

  To start SAS Forecast Studio, click **Launch**.

- Use the direct link to the Java Web Start client. You can launch SAS Forecast Studio from **`http://`*`your-server-name`*`:8080/SASForecastServer/main.jnlp`**.

Configuring the Logon Behavior for Java Web Start

Default Logon Behavior for the Desktop Application and Java Web Start

The overall behavior of SAS Forecast Studio is the same whether you launch SAS Forecast Studio from your desktop or from Java Web Start. The main difference between the two is visible in the Log On dialog box.

- When you open SAS Forecast Studio from your desktop, the **SAS environment** drop-down list defaults to the last environment that you used on that machine. If you have not previously logged on to SAS Forecast Studio from this machine, the **SAS environment** drop-down list displays the default environment.
- If you launch SAS Forecast Studio using the Java Web Start client, the **SAS environment** drop-down list includes the **(host deployment)** option in addition to the available SAS environments. This option represents the server from where SAS Forecast Studio was downloaded. When you open SAS Forecast Studio using the Java Web Start client, the **SAS environment** drop-down list defaults to the last environment that you used. If you have not previously logged on to SAS Forecast Studio, the **SAS environment** drop-down list defaults to the **(host deployment)** option.

Configuring the Logon Behavior on the Server

Unlike the desktop application, the Java Web Start client often does not have a pre-configured list of SAS environments that appear in the **SAS environment** drop-down list. In many cases, the **(host deployment)** option might be the only one available. You can configure this list of environments in several ways.

For simple configurations of SAS Forecast Server, it might be sufficient to simply specify these configuration properties for the Java Web Start client:

- the URL to the file that specifies the SAS environment options. For more information, see "How to Specify the Value of the SAS Environment URL" on page 56.
- the environment that should be selected by default when the Log On dialog box appears. You can specify the preferred environment in SAS Management Console. For more information, see "Set the Preferred Environment in the Log On Dialog Box for SAS Forecast Studio" on page 57.
- whether users have permission to select a different environment in the Log On dialog box. You can prevent users from changing the environment by using the **Lock SAS environment selection** property in SAS Management Console. For more information, see "How to Specify the Value of the SAS Environment URL" on page 56.

The disadvantage to configuring the logon behavior using these configuration properties is that the logon behavior depends only on the server where the SAS Forecast Studio client was downloaded from, so all machines and users who rely on the Java Web Start client must use the same settings.

Configuring the Logon Behavior on the Client

On the client machine, you can configure the logon behavior in either of these ways:

- Specify the URL for the sas-environment.xml file in the SAS_ENV_DEFINITION_LOCATION environment variable.
- If you ran the SAS Deployment Wizard on that machine and the URL for the sas-environment.xml file was configured during deployment, create the SASHOME environment variable. When this environment variable is defined, the desktop client automatically checks the local configuration files for a configured sas-environment.xml file to use.

If you configure the logon behavior in both ways, then the value from the SAS_ENV_DEFINITION_LOCATION environment variable is used.

Here is how you can create either of these environment variables on a machine running Windows XP. If you are running another version of Windows, then these steps might be slightly different for your operating environment.

1. On your desktop, right-click **My Computer** and select **Properties**. The System Properties dialog box appears.
2. Click the **Advanced** tab and click **Environment Variables**. The Environment Variables dialog box appears.
3. Click **New** to create a new system variable.

Environment Variable to Create	Variable Type	Variable Value
SAS_ENV_DEFINITION_LOCAION	**`SAS_ENV_DEFINTION_LOCATION`**	The location of the sas-environment.xml file. An example of this path is **`http://Web-server-name/server-path/sas-environment.xml`**.

Environment Variable to Create	Variable Type	Variable Value
SASHOME	**SASHOME**	The location where SAS was installed. An example of this path is **C:\ \Program Files \SASHome**.

Click **OK** to close the New System Variable dialog box. SAS_ENV_DEFINITION_LOCATION or SASHOME now appears in the list of system variables.

4. Click **OK**.

Note: When you open a new Web browser, the process for that browser uses the cached values of the environment variables. Because Java Web Start is a child of this process, the Java Web Start client also uses these cached values. In order for Java Web Start to recognize the new environment variables (or any changes that you make to these environment variables), you must restart your Web browser and then reopen the Java Web Start client.

Starting SAS Forecast Studio with Options

Specify the Number of Observations to Use to Detect the Time Interval

You can specify the number of observations from the input data set that SAS Forecast Studio uses to detect the time interval. By default, SAS Forecast Studio uses the first 10,000 observations to detect the time interval. Usually, a few distinct time ID values are needed to detect the interval. However, data sets with many repeated values of the time ID variable sometimes require a larger sample to get enough distinct values. Increasing the number of observations might improve time interval detection when you are using data sets that contain many repeated values. Values of 100,000 and larger could cause noticeably slower performance when you are creating a new project.

To configure SAS Forecast Studio to use a customized time interval value:

1. Open the forecaststdo.ini file in a text editor. For example, in a default Windows installation, this file is located in the **C:\Program Files\SAS \SASForecastStudio\4.1** directory.
2. In the .ini file, type a new JavaArgs_*n* entry, where *n* is the number of the next argument in the list. Increment the argument number accordingly. For example, type the following:
```
JavaArgs_14=-Dcom.sas.analytics.forecasting.interval_
sample=50000
```

Specify the Sample Size for Validating BY Variable Values and Formats

When you start SAS Forecast Studio, you can choose an option that specifies a default value to use for checking the validity of BY variable values and formats. This option applies when you use the new project wizard to save a project's code without running it.

The default value for the sample size is 50000. When you specify a value for this property, the first *n* observations of the data set are used to check for the following:

- Is any numeric BY variable continuous-valued?
- Does any BY variable have a format that maps more than one raw value to the same formatted value?

If the answer to either question is yes, then an appropriate error dialog box is displayed, and the project is not created or saved.

To configure SAS Forecast Studio to use a sample size other than 50000 to validate BY variable values and formats:

1. Open the forecaststdo.ini file in a text editor. For example, in a default Windows installation, this file is located in the **`C:\Program Files\SAS \SASForecastStudio\4.1`** directory.
2. In the .ini file, type a new JavaArgs_*n* entry, where *n* is the number of the next argument in the list. Increment the argument number accordingly. For example, type the following:
   `JavaArgs_14=-Dcom.sas.analytics.forecasting.by_validation_ sample=75000`

Chapter 9

Security Administration Tasks

Enabling Users to Access SAS Forecast Server

Authentication versus Authorization

Before learning about authorizations within SAS Forecast Server, you first need to understand the difference between user authentication and user authorization.

- User authentication is an identity verification process that attempts to determine whether users are who they say they are.
- User authorization is the process of determining which authenticated users have which permissions for which resources and actions. User authorization happens after user authentication.

For more information about the SAS security scheme, see the security section of the SAS Intelligence Platform documentation at **http://support.sas.com/documentation**.

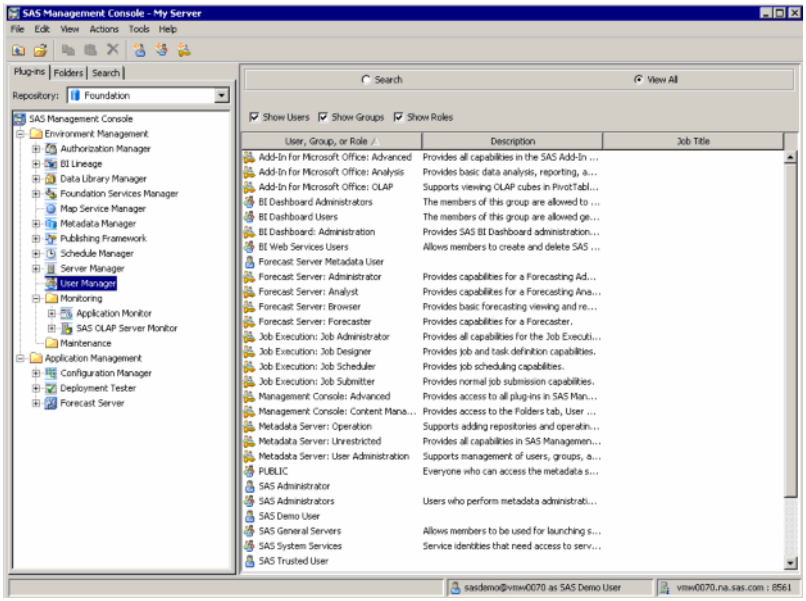
Initial Authentication

Authentication is the verification of your site identity based on the provided credentials. You can then use that site identity to verify that you have a working metadata identity in the SAS Intelligence Platform. To be authenticated, a user must have a valid account for their site's network and an associated metadata account for the SAS Metadata Server. For more information, see the *SAS Intelligence Platform: Security Administration Guide*.

Initial Users

When you install and configure the SAS Intelligence Platform and SAS Forecast Server, standard user and group definitions are added. The following display shows an example of how these users and groups might appear in the User Manager plug-in of SAS Management Console.

Display 9.1 *Example of Standard Users and Groups Created for SAS Forecast Server*



You can also define new users or groups. For more information, see the SAS Management Console Help.

Understanding the SAS Forecast Server Metadata User

What Is the SAS Forecast Server Metadata User?

To access data that is not accessible to the current user and to update the metadata, SAS Forecast Server uses the SAS Forecast Server Metadata User (fsmeta@saspw). This account must have access to all metadata that is used by the product. For example, the WriteMetadata permission is required for any object that the product might need to modify, and the ReadMetadata permission is required for any object that the product might need to use or detect.

During deployment, the SAS Forecast Server Metadata User is granted permissions to the **`/System/Applications/SAS Forecast Server`** metadata folder. This folder contains the metadata objects that represent the product content.

If you create additional metadata folders (folders that are outside of the **`/System/Applications/SAS Forecast Server`** directory) that you want to use with SAS Forecast Server, you must grant the Forecast Server Metadata User the appropriate permissions on these folders. Additional metadata folders are created to manage the report metadata objects. Special metadata objects that are used by the product (such as objects that define libraries and servers) must also be accessible. Finally, the permission settings must be sufficient to ensure that the ReadMetadata permission is granted on all other objects that are created during the deployment.

Note: A product environment cannot be associated with a server without having WriteMetadata permission on the defining server object. This security requirement is imposed by the SAS Metadata Server.

Specify User Permissions for the Forecast Server Metadata User

To grant permissions to a folder outside of the **`/System/Applications/SAS Forecast Server`** directory:

1. Open SAS Management Console, and log on as the SAS Administrator (for example, sasadm@saspw).
2. Click the **Folders** tab.
3. Locate the metadata folder that you created. For example, you might create a **`/Shared Data/Forecast Server 4.1/Reports`** folder to organize the reports that you use in the product.
4. Right-click on the metadata folder, and then select **Properties**. The Properties dialog box appears.
5. Click the **Authorization** tab.
6. In the **Users and Groups** area, click **Add**. The Add Users and Groups dialog box appears.
7. In the **Available Identities** list, double-click on the **Forecast Server Metadata User**. The Forecast Server Metadata User is added to the **Selected Identities** list. Click **OK**.
8. In the **Users and Groups** area, select **Forecast Server Metadata User**.
9. Grant the **ReadMetadata**, **WriteMetadata**, **WriteMemberMetadata**, and **CheckinMetadata** permissions for the Forecast Server Metadata User, and then click **OK**.

TIP When appropriate, grant the ReadMetadata, WriteMetadata, WriteMemberMetadata, and CheckinMetadata permissions on the parent folder. Any child folders automatically inherit these permissions from the parent folder.

Project Owners

What Is a Project Owner?

The user who creates a project in SAS Forecast Studio is the de facto administrator for that project. Such users are referred to as project owners. In addition to the normal project capabilities, project owners can use management actions, such as the ability to change project-sharing settings and the ability to delete the project. You can transfer project ownership to other users by using either the SAS Forecast Server Plug-in for SAS Management Console or the FSSETOWN macro. A project can have only one owner.

The value for the project owner is detected by comparing a generated identity token with a previously stored token value for the project. To determine the value for the project owner, either look at the SAS Forecast Studio Status page or the **Show only my *token-value* projects** check box in the Projects dialog box in SAS Forecast Studio.

How to Change Project Ownership

To change project ownership using the SAS Forecast Server Plug-in for SAS Management Console:

1. Open SAS Management Console. To log on, your user account must be assigned to the Forecast Server: Administrator group. You must also have a valid operating system account.

   Note: Do not log in as sasadm@saspw because this account should not be configured as a product user.

2. Select the **Forecast Server** node on the **Plug-ins** tab.
3. Expand the environment node that contains the project. Then expand the **Projects** node.
4. Right-click on the desired project, and then select **Properties**. The Project Properties dialog box appears.
5. Type the new user name in the **Owner** text box, and then click **OK**.

Note: You can also use the Access Settings batch operation to set the same owner on multiple projects. This feature is available in the SAS Forecast Server Plug-in for SAS Management Console. Right-click on the environment node, and select **Batch Operations ⇨ Access Settings**. For more information, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Securing Access to SAS Forecast Server

Security Layers

Security settings in SAS Forecast Server are implemented in four layers:

- Capabilities – enforced by the product
- Metadata Permissions
  - ReadMetadata, WriteMetadata, WriteMemberMetadata, CheckinMetadata – enforced by the SAS Metadata Server
  - other permissions (such as library authorization) – enforced by the product
- Project Ownership and Sharing (also called the application security model) – enforced by the product
- File System Permissions – enforced by the operating system

Typically, permissions are checked in the order listed. For example, capabilities are checked first while file system permissions are checked last.

Notes Regarding Permissions

File System Permissions

Use the metadata and application layers when you define your security strategy. Typically, file system permissions are not checked before an operation is attempted in the product.

Note: Securing content only through the use of file system permissions is not recommended.

WriteMetadata Permissions

In SAS Forecast Server, the metadata security that is applied to the objects with product content is limited to controlling the visibility of this content by using the ReadMetadata permission. Metadata updates to these objects are performed indirectly by the SAS Forecast Server Metadata User. The WriteMetadata settings for individual user accounts have no impact on the product behavior, so the WriteMetadata permission should be used on objects that contain product content.

What Permissions Can You Control by Using Metadata?

To secure access to metadata objects that represent the SAS Forecast Server data, you can grant or deny permissions to individuals or groups by using the **Authorization** tab in SAS Management Console.

The following table lists some of the metadata objects and their permissions.

Metadata Object	Permissions
environment	ReadMetadata – controls visibility
projects	ReadMetadata – controls visibility
libraries	ReadMetadata – controls visibility Read – controls ability to read contents Write – controls ability to write contents
tables that are registered to libraries	Read – controls ability to read contents Write – controls ability to write contents
columns in registered tables	Read – controls ability to read contents Write – controls ability to write contents
reports (stored process)	ReadMetadata – controls visibility

Note: Other permissions, such as the ability to use ReadMetadata to hide server objects, can impact product behavior. However, these permissions are usually not needed under typical operating conditions.

Environment Permissions

What Are Environments?

An environment enables you to group projects together into workspaces. It is defined by the following basic attributes:

- logical name
- host workspace server
- content path on the server

An environment defines each of the following:

- a container for projects
- a workspace for users
- a partition of a workspace server

An environment is implemented as a metadata folder, metadata object, and directory structure (in the server's file system). Only metadata and file system permissions apply to these structures. To enable project sharing for an environment, you must configure the directory tree that is associated with the environment full control to all users that are allowed to access the environment. Note that this configuration relies on the metadata settings to secure the environment.

In addition, setting the ReadMetadata permission at the environment level controls whether a user can access the projects and reports associated with that environment. When a user logs on to SAS Forecast Studio, environments for which a user does not have ReadMetadata permission do not appear as options in the Log On dialog box. Therefore, the user cannot select this environment and use its content in any SAS Forecast Server client.

Note: Each file system location in a workspace server should be used by only one environment system-wide, regardless of the number of environments or middle tiers in the system. In general, the file system location that is specified for the environment and its subdirectories should be reserved for that environment only. This restriction ensures the integrity of the file system content and the ability to properly coordinate accesses by clients.

You can create environments by using the FSNEWENV macro or by using the SAS Forecast Server Plug-in for SAS Management Console. For more information about using the plug-in to create an environment, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Configure Environment Permissions

To configure permissions for an environment:

1. In the file system, grant full control on the environment directory to all environment users or user groups. For easier management, a user group is recommended. For more information about user groups, see "Creating Operating System Accounts in UNIX Environments" on page 37.
2. Open SAS Management Console, and log on as the SAS Administrator (for example, sasadm@saspw).
3. Click the **Folders** tab.
4. Expand the **System**, **Applications**, **SAS Forecast Server**, **SAS Forecast Server 4.1**, and **Environments** folders.
5. Right-click on the environment folder and select **Properties**. The environment Properties dialog box appears.
6. Click the **Authorization** tab.
7. Select your user (or a group of users) in the **Users and Groups** area.
8. Set the **ReadMetadata** permission for your user or group and then click **OK**.

TIP You can also set the permissions for an environment directly on the environment object in the environment folder.

Project Permissions

About Project Permissions

Projects are similar to environments in physical structure, with file content being saved to an environment subdirectory. By default, the security for the file system is specified when you create an environment. No additional security is required when projects are created. However, you can set the ReadMetadata permission on each project object to control that project's visibility to users.

Configure Project Permissions

To configure the permissions for a project metadata object:

1. Open SAS Management Console, and log on as the SAS Administrator (for example, sasadm).
2. Click the **Folders** tab.
3. Expand the **System**, **Applications**, **SAS Forecast Server**, **SAS Forecast Server 4.1**, and **Environments** folders.
4. Select the environment folder.
5. Right-click on a project, and then select **Properties**. The *project* Properties dialog box appears.
6. Click the **Authorization** tab.
7. Select your user (or a group of users) in the **Users and Groups** area.
8. Set the **ReadMetadata** permission for your user or group, and then click **OK**.

Sharing Options for Projects

About Sharing Projects

By default, only the project owner or a SAS Forecast Server administrator can open a given project. By enabling project sharing, any user who can see a project can access the project. One way to enable sharing is to select the **Allow other users to view and edit this project** check box in the New Project wizard or the Project Properties dialog box.

Note: For migrated projects, you cannot change the sharing status of the project until the project is compatible with the current version of SAS Forecast Server. You can use the update batch operation to update multiple projects. In the SAS Forecast Server Plug-in for SAS Management Console, right-click on the environment node, and select **Batch Operations** ⇨ **Update**. For more information, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.

Configure Sharing for a Project

To enable sharing for a project in the SAS Forecast Server Plug-in for SAS Management Console:

1. Open SAS Management Console. To log on, your user account must be assigned to the Forecast Server: Administrator group. You must also have a valid operating system account.
2. Expand the **Forecast Server** and an environment node.

3. Right-click on a project and then select **Properties**. The project Properties dialog box appears.
4. Select **Allow other users to view and edit this project**, and then click **OK**.

For more information about how to enable project sharing in SAS Forecast Studio, see the *SAS Forecast Studio: User's Guide*.

Sharing and Groups

Although project ownership and the sharing security model do not include the explicit concept of a user group, the management structure does divide users into implied groups. For example, all users who can access a particular environment can be considered a group. Therefore, when you enable sharing on a project, you are essentially sharing the project with other members of your group. A user who does not have access to the environment where your project is stored still cannot access the project. You should use caution when copying projects that are shared to another environment because the scope of the access that is allowed by the sharing option depends on how the related environment is configured.

Chapter 10

Report Administration Tasks

Overview of Reports

What Is a Report?

In SAS Forecast Server, reports provide a mechanism for you to extend the capabilities of the product so that you can perform site-specific custom operations. Reports encapsulate custom logic and when the report is executed, it receives information about the run-time state in SAS Forecast Studio. By collecting information about the state of the project, SAS Forecast Server knows what the user is looking at, and consequently what data to use when generating the report.

A report is written in the SAS programming language and is saved as a SAS Stored Process. A stored process is a SAS program that is stored centrally on a server. Stored processes consist of two distinct parts: the SAS code and the stored process definition that resides on a metadata server. A client application can then execute the program, and can receive and process the results. Stored processes enable you to maintain and manage code centrally, give you better control over changes, enhance security and application integrity, and ensure that every client executes the latest version of code that is available.

Stored processes are like other SAS programs, except that they have an additional feature that enables you to customize the program's execution. This feature enables the invoking application to supply parameters at the time that the stored process is invoked. For example, if you have a stored process that analyzes monthly sales data, you could create a MONTH variable in the stored process. At execution time, you would supply

the parameter MONTH=MAY to analyze May sales data. For more information about how to create a stored process and to invoke it in a client application, see the *SAS Stored Processes: Developer's Guide* at **http://support.sas.com/documentation/onlinedoc/inttech**.

Reports are implemented using stored processes. However, not all SAS stored processes generate reports. The reports in SAS Forecast Server differ from generic stored processes in these ways:

- The reports rely on a special infrastructure that includes the macros that are delivered with SAS Forecast Server. As a result, reports reflect the run-time state of SAS Forecast Studio.
- SAS Forecast Server includes a custom execution mechanism for reports. In fact, most reports work only when run through this mechanism. These reports are not intended to be run outside of SAS Forecast Server.

To distinguish reports from other stored processes, SAS Forecast Server identifies them by the FS_REPORT keyword.

Types of Reports

In SAS Forecast Server, you can have two types of reports.

- Several sample reports that demonstrate how reports can be used to handle a variety of tasks ship with SAS Forecast Server. These sample reports can also serve as a basis for a custom report. Simply copy the sample report and then modify the copy to meet your site's needs.

  The reports are not available for use in the Reports and Stored Processes dialog box until you deploy them using the SAS Forecast Server Plug-in for SAS Management Console. For a list and description of the sample reports, see the *SAS Forecast Studio: User's Guide*.

- To meet the needs of your site, you can also create custom reports.

  While working with custom reports in SAS Forecast Server, note these constraints:

  - SAS Forecast Server does not currently support the execution of reports using a SAS Stored Process Server.
  - For stored processes that are compatible with SAS 9.2, each metadata object is configured to use a specific execution server. When you use multiple SAS Workspace Servers to host projects, you might need to create one metadata object for each SAS Workspace Server. Note that you must also copy the source file for the stored process to the file system of each SAS Workspace Server. In this way, each metadata object can refer to the source file that is located in its file system.
  - The Reports and Stored Processes dialog box in SAS Forecast Studio lists only reports that run on the same SAS Workspace Server as the current project. Reports present in metadata but not shown in this dialog box are either not visible to the user (for example, the ReadMetadata permission is not granted or the Use Reports capability has not been assigned to the user), missing the FS_REPORT keyword, or configured to use a different server.

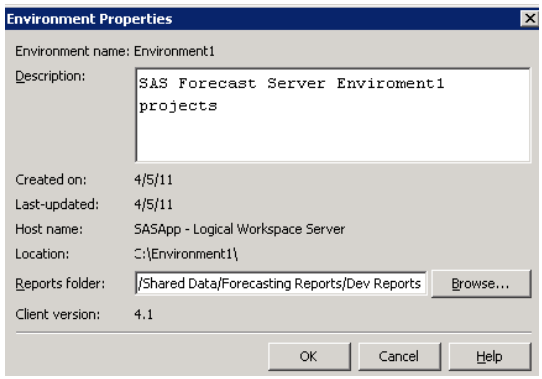
When you register a report object in SAS Management Console for use with the product, you must use the FS_REPORT keyword to distinguish the report object from other stored processes. In addition, the sample reports provided with the product use the FS_SAMPLE keyword to distinguish the sample reports provided with the product from custom user reports.

Managing Reports

To use reports in SAS Forecast Studio, you must first create a report metadata folder and configure your environment to use this metadata folder location. The existence of the report folder enables the Reports and Stored Processes dialog box in SAS Forecast Studio.

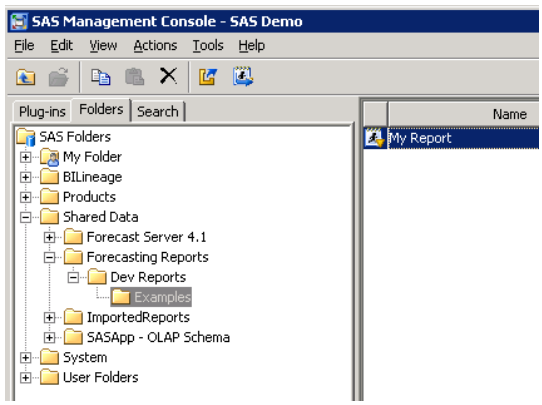
For example, suppose that you configure your environment to use the **`/Shared Data/Forecasting Reports/Dev Reports`** metadata folder as shown in the following display.

Display 10.1 *Configuring the Environment's Root Report Metadata Folder*



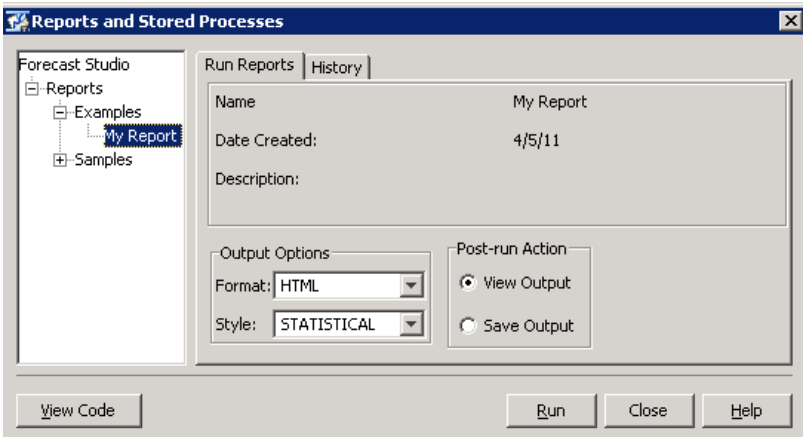
In addition, suppose that you create the **My Report** stored process metadata object at the metadata folder location **`/Shared Data/Forecasting Reports/Dev Reports/Examples`** as shown in the following display.

Display 10.2 *Report Metadata Subfolder*



In this example, **My Report** is displayed under the **`Reports/Examples`** node in the Reports and Stored Processes dialog box.

Display 10.3 *Folder Presentation in the Reports and Stored Processes Dialog Box*



The **Reports** node at the top of the hierarchy represents the root folder that was configured in the metadata. This path does not appear in the Reports and Stored Processes dialog box.

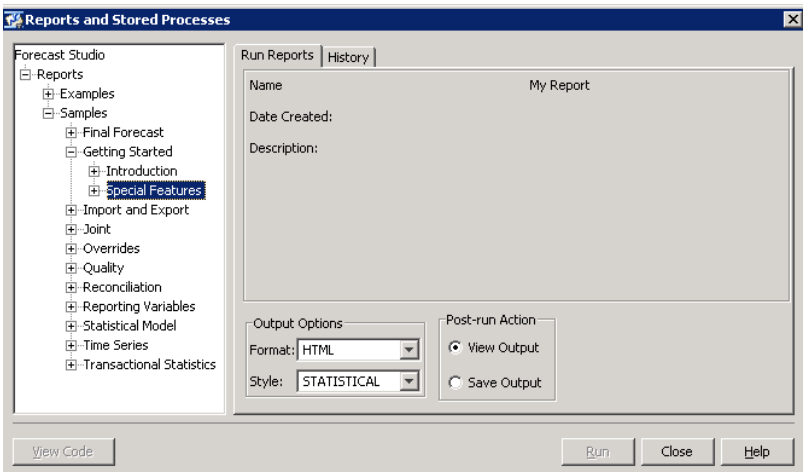
Before you can create a report for use in SAS Forecast Studio, you must let the environment know where your reports metadata folder is located. You can configure the location for the reports metadata folder in the environment's properties dialog box.

Accessing the Reports in SAS Forecast Studio

You can access and execute reports when you have a project open in SAS Forecast Studio by selecting **Tools** ⇨ **Reports and Stored Processes**.

Here is an example of the Reports and Stored Processes dialog box in SAS Forecast Studio.

Display 10.4 *SAS Forecast Studio Reports and Stored Processes Dialog Box*



The reports shown in SAS Forecast Studio are filtered based on whether the user is assigned the Use reports capability, whether the ReadMetadata permission has been specified for a report, use of the FS_REPORT keyword, and the server constraints that

you specified in the metadata. To use a report, it must be on the same server as the project. When you run a report, the locations of the data and forecasts for the current project are passed to the stored process.

Using the Sample Reports

Deploy the Sample Reports

To deploy the sample reports:

1. Create a metadata folder to store your report objects.
2. Grant all permissions on the folder to the SAS Forecast Server Metadata User.
3. Configure the user permissions on the report folder.
4. Designate a root report folder for the environment.
5. Select the SAS Management Console **Plug-ins** tab.
6. Expand the **Forecast Server** node.
7. Expand the environment node where you want to deploy the sample reports.
8. Right-click the **Reports** node, and select **Deploy Samples**.

Remove the Sample Reports

To remove all sample reports in the current environment's report tree:

1. In SAS Management Console, click the **Plug-ins** tab.
2. Expand the **Forecast Server** node.
3. Expand the environment node where you want to remove the default reports.
4. Right-click the **Reports** node, and select **Remove Samples**.
5. (Optional) To remove an empty metadata folder from the report tree, right-click the **Reports** node, and select **Prune**.

   Note: The **Reports** node does not appear under an environment node until the report tree is configured.

CAUTION:

Using the Remove Samples option removes any report object with the FS_SAMPLE keyword. Before using the Remove Samples option, you should remove the FS_SAMPLE keyword from any custom reports that were created by copying a sample report.

Registering Reports

Preregistration Tasks

The following tasks enable you to gather the information that you need to make a report usable in SAS Forecast Studio.

To prepare for report registration:

1. Locate the source code for the report. The source code can be saved to a file system or in a metadata repository.

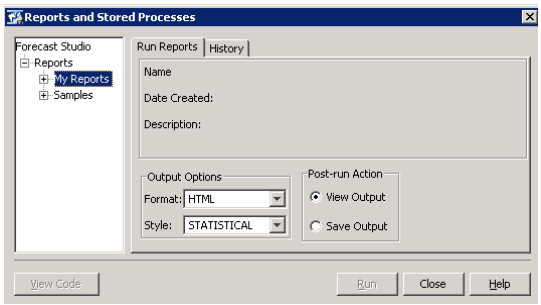
   For example, the example.sas file is saved in the `C:\MyStoredProcesses` directory. The source repository location is `C:\MyStoredProcesses`, and the source filename is example.sas.

   Note: The file system location must be on the same physical machine that hosts the corresponding project files for SAS Forecast Server. Having these files on the same machine ensures that the project data is accessible to the stored process when it executes.

2. Plan the organization of your report hierarchy. If you want to display the report under its own folder group in the Reports and Stored Processes dialog box in SAS Forecast Studio, then you can create this hierarchy in the metadata using SAS Management Console. The Reports node in the dialog box corresponds to the reports root folder that you configured for that environment in SAS Management Console. The expandable nodes in the dialog box correspond to subfolders that you create under the root folder. For example, you can specify that a **My Reports** folder is displayed in the Reports and Stored Processes dialog box under the **Reports** tree by creating a subfolder with that name within the root folder.

Here is how such a custom folder structure would appear.

Display 10.5 *Example Report Hierarchy*

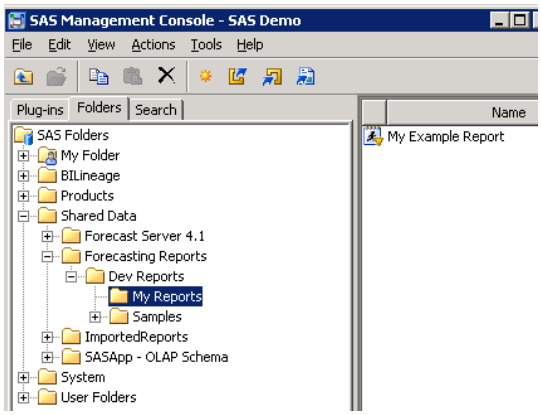


For more information about creating metadata folders, see the documentation for SAS Management Console at `http://support.sas.com/documentation/onlinedoc/sasmc`.

3. Select a name for the report. The report name that you specify in SAS Management Console is the metadata object name. This report name is displayed in the Reports and Stored Processes dialog box in SAS Forecast Studio. For example, if you name the report **My Example Report** in SAS Management Console, then the Reports and Stored Processes dialog box displays the name, **My Example Report**.

Here is an example folder structure that uses the **My Reports** subfolder that contains the **My Example Report**.

Display 10.6 *Report Hierarchy with an Example Report*



Register a Report

For information about the basic processes and terminology that are related to using stored processes, see the *SAS Stored Processes: Developer's Guide* at **http://support.sas.com/documentation/onlinedoc/inttech**.

Note: Before you alter any existing stored processes, it is recommended that you make a copy of the stored process metadata object and its associated source code.

Before you register a report, complete these steps in SAS Management Console:

1. Create a metadata folder to store your report objects.
2. Write the code for the report.
3. Set permissions on the report metadata folder so that users can access it in SAS Forecast Studio.
4. Configure your environment to use the reports metadata folder.
5. Create a stored process metadata object.
   a. Open SAS Management Console and log on as an administrator (for example, sasadm).
   b. Click the **Folders** tab.
   c. In the root directory for your reports, right-click and select **New Stored Process**. The New Stored Process wizard appears.
   d. In the General step, type the name of your SAS program, **`My Example Report`**, and an optional description.
   e. In the **Keywords** section, click **Add**. The Add Keyword dialog box appears.
   f. Type **`FS_REPORT`**, and then click **OK**. Click **Next**.
   g. In the Execution step, set the **Application server** option to the SAS Workspace Server that hosts the project files.

      Note: The execution constraints for a stored process must enable you to run the stored process on the SAS Workspace Server that hosts the projects that use this report.

h. For the **Source code repository** option, select the directory where you saved your SAS code (in this example,`C:\MyStoredProcesses`). If your directory does not appear in the list of options, you can add a new location.

i. For the source file, specify `example.sas`, and for the output type, select **Package**. Click **Next**.

Note: Starting in SAS 9,3, you can save the source code with the stored process metadata. By saving the code with the metadata, you can run the stored process on multiple application servers. When the source code is saved to a directory on the server, you can run the stored process only on the server where the source code is saved.

j. Click **Finish** in the Parameters panel.

Note: For more information about defining parameters, see the *SAS Stored Processes: Developer's Guide* at `http://support.sas.com/documentation/onlinedoc/inttech`.

Creating Reports

Types of Report Parameters

What Is a Normal Report Parameter?

A normal report parameter is a stored process prompt that is completely defined in the metadata. Examples of normal parameters include specifying the title to use for the output and specifying whether to include the observation number in the results. These parameters are created through the usual interfaces for defining stored process prompts (for example, SAS Management Console). The only constraint is that the names of normal report parameters cannot start with the dynamic parameter prefixes. (See Table 10.1 on page 82.)

What Is a Dynamic Report Parameter?

A dynamic report parameter is one for which you select from options that are provided by the product at run time. Because the options depend on the run-time state, they cannot be defined in the metadata ahead of time. For example, a parameter that enables you to select an independent variable from the current project is a dynamic parameter.

Dynamic parameters require special support from the product to generate the list of available options. They are not extensible. To find dynamic parameters in a report, the product checks for a special prefix in the name of the text (type) parameter that matches a prefix in the following table. If a match is found, then the corresponding options list is generated and assigned to the parameter before the report prompt is displayed. If a match is not found, then the product assumes that the parameter is a normal parameter and uses it as is.

Table 10.1 *Dynamic Parameter Prefixes*

Parameter Name	Elements in Options List
FS_BYVAR	All project BY variables

Parameter Name	Elements in Options List
FS_CURRENT_BYVAR	BY variables in the selected hierarchy level
FS_DEPVAR	All project dependent variables
FS_INDVAR	All project independent variables
FS_REPORTVAR	All project reporting variables
FS_EVENT	All project events
FS_FITSTAT	Statistics of fit
FS_MODELSTAT	All model statistics
FS_LIBRARY	Libraries allowing Read access
FS_WLIBRARY	Libraries allowing Write access
FS_OLAPSCHEMA	All OLAP schemas currently defined in metadata

When locating dynamic parameters, the product does the following:

1. Prefix matching allows multiple variables to use the same dynamic list. For example, FS_INDVAR1 and FS_INDVAR2 both match FS_INDVAR. Therefore, an independent variable options list would be provided.
2. For efficiency, only prefixes generated by truncating the name at non-letter characters are considered for prefix matching. For example, FS_INDVARS does not match the prefix FS_INDVAR because the name breaks on the letter S. However, the prefixes FS_INDVAR, FS_INDVAR_LIST, and FS_INDVAR1 all match the FS_INDVAR prefix.
3. If a required parameter has an empty options list, the product detects this and does not execute the report. This prevents the issue of a prompt that cannot be satisfied. For example, you cannot use a required parameter named FS_INDVAR with a project that does not contain independent variables.

What Is an Internal Report Parameter?

To assist you with creating stored processes, the product provides internal parameters, also called predefined macro variables. The internal parameters are automatically defined and populated by the product and do not appear in the metadata definition of the report. These parameters are used to pass state information and instructions to the stored process environment. In particular, they convey information about the project and the location in the hierarchy that you are currently viewing.

Note: To ensure correct behavior, user-defined parameters should not use the same names as internal parameters. The naming convention used for the sample reports, and generally recommended, is to prefix metadata parameters with **`FS_`**. This prevents name collisions since all internal parameters have the **`HPF_`** prefix.

By default, the maximum length for a macro variable is 4,096. However, if the number of variables (BY, dependent, independent, reporting, and so on) is very large, you can

increase the maximum length for the macro variable to 65,534 by using the MVARSIZE= system option.

Note: In the following tables that describe the macro variables, the variable names that end with # indicate a sequenced variable. Unless otherwise stated, you can assume that sequencing begins with 1 for these variables. For levels, indexing starts at the top level.

Control Macro Variables

The following variables are used for basic session control.

Table 10.2 *Control Macro Variables*

Name	Description	Format
HPF_READ_ONLY	Project access Read-Only flag By default, stored processes have Read-Only access with respect to the project libraries. This macro variable changes this access to Write. For example, the following SAS code assigns project library names with Read-Only access: `%include "&HPF_INCLUDE";`	Boolean
HPF_ODSDEST	Used inside a stored process to control the output destination for the report. By default, HPF_ODSDEST is set to HTML, which means that the output is in HTML format.	ODS destinations
HPF_ODSSTYLE	Used inside a stored process to control the output style.	ODS styles
HPF_MULTISELECT_VAR_COUNT	Used to specify the number of multi-value metadata parameters passed to the report.	Positive integer
HPF_MULTISELECT_VAR#	Used to specify the base name of each multi-value metadata parameter.	Multi-value metadata parameter name

Metadata Macro Variables

The following variables describe the metadata resources that are used by SAS Forecast Server.

Table 10.3 Metadata Macro Variables

Name	Description	Format
HPF_METADATA_HOST	Metadata server host name	Host name
HPF_METADATA_PORT	Metadata server port number	Port number
HPF_METADATA_ REPNAME	Metadata repository name	Repository name

Library Macro Variables

The following variables describe the library resources that are used by SAS Forecast Server.

Table 10.4 Metadata Macro Variables

Name	Description	Format
HPF_ENV_LIBNAME_ COUNT	Number of local environment libraries	Positive integer
HPF_ENV_LIBNAME#	LIBNAME of each local environment library	LIBNAME library names
HPF_ENV_LIBPATH#	File system path of each local environment library	Directory path
HPF_ENV_LIBACCESS#	Access level of each local environment library	Positive integer
HPF_META_LIBNAME_ COUNT	Number of manually assigned metadata libraries	Positive integer
HPF_META_LIBNAME#	LIBNAME of each manually assigned library	LIBNAME library names
HPF_META_LIBID#	FQID of the metadata object that defines each manually assigned library	Metadata FQID value
HPF_META_LIBACCESS#	Access level of each manually assigned library	Positive integer

Basic Project Macro Variables

The following variables describe the basic characteristics for the active project.

Table 10.5 *Basic Project Macro Variables*

Name	Description	Format
HPF_PROJECT	The project name.	SAS name
HPF_DESC	The project description.	SAS label
HPF_PROJECT_SERVER	The logical name of the host SAS Workspace Server.	Host name
HPF_DEFAULT_ LOCATION	The base file system path to the environment directory.	File system path
HPF_PROJECT_LOCATION	File system path to the project directory.	System path
HPF_INCLUDE	The file system path to the project include file. This macro variable specifies the system path and filename that contains the SAS code to assign the SAS libraries and catalogs that are associated with the project. By default, all libraries and catalogs are assigned with Read-Only access (ACCESS=READONLY). For example, the following SAS code assigns project library names with Read-Only access: `%include "&HPF_INCLUDE";`	System filename
HPF_EVENTS	Space-delimited list of all event names.	SAS name
HPF_NUM_EVENTS	Number of events defined in the project.	Positive integer
HPF_EVENT_#	The name of individual event elements.	SAS name

Data Hierarchy and Settings Macro Variables

The following variables describe the data settings for the active project.

Table 10.6 *Data Hierarchy and Settings Macro Variables*

Name	Description	Format
HPF_INPUT_LIBNAME	SAS library reference where the source data for the input data set was obtained.	SAS name
HPF_INPUT_DATASET	Data set from where the source data was obtained.	SAS name
HPF_NUM_BYVARS	Number of BY variables. If there are no BY variables, HPF_NUM_BYVARS is set to zero.	Positive integer
HPF_BYVAR#	Individual BY variable names listed in the n^{th} position of the ordered list of BY variables (HPF_BYVARS). The first BY variable name is stored in HPF_BYVAR1, the second in HPF_BYVAR2, and the last is stored in HPF_BYVAR&HPF_NUM_BYVARS. If there are no BY variables (&HPF_NUM_BYVARS is zero), these macro variables are not defined.	SAS name
HPF_NUM_DEPVARS	Number of dependent variables. There is always at least one dependent variable.	Positive integer
HPF_DEPVARS	Space-delimited list of all dependent variable names. The order of the dependent variable names is the same as specified in the project.	List of SAS names separated by a single space
HPF_DEPVAR#	Individual dependent variable names listed in the n^{th} position of the ordered list of dependent variables (HPF_DEPVARS). The first dependent variable name is stored in HPF_DEPVAR1, the second in HPF_DEPVAR2, and the last is stored in HPF_DEPVAR&HPF_NUM_DEPVARS. Since there is always at least one dependent variable associated with a project, HPF_DEPVAR1 is always defined.	SAS name
HPF_NUM_INDEPVARS	Number of independent variables. If there are no independent variables, HPF_NUM_INDEPVARS is set to zero.	Positive integer

Name	Description	Format
HPF_INDEPVARS	Space-delimited list of all independent variable names. The order of the independent variable names is the same as specified in the project. The macro variable is always defined; but if there are no independent variables, HPF_INDVARS is set to NULL.	List of SAS names separated by a single space
HPF_INDEPVAR#	Individual independent variable names listed in the n^{th} position of the ordered list of independent variables (HPF_INDEPVARS). The first independent variable name is stored in HPF_INDEPVAR1, the second in HPF_INDEPVAR2, and the last is stored in HPF_INDEPVAR&HPF_NUM_INDEPVARS. If there are no independent variables (&HPF_NUM_INDVARS is zero), these macro variables are not defined.	SAS name
HPF_NUM_REPORTVARS	Number of reporting variables. If there are no reporting variables, then HPF_NUM_REPORTVARS is set to zero.	Positive integer
HPF_REPORTVARS	Space-delimited list of all reporting variable names. The order of the reporting variable names is the same order as specified in the project. The macro variable is always defined; but if there are no reporting variables, then HPF_REPORTVARS is set to NULL.	List of SAS names separated by a single space
HPF_REPORTVAR#	Individual report variable names listed in the n^{th} position of the ordered list of report variables (HPF_REPORTVARS). The first report variable name is stored in HPF_REPORTVAR1, the second in HPF_REPORTVAR2, and the last is stored in HPF_REPORTVAR&HPF_NUM_REPORTVARS. If there are no reporting variables (&HPF_NUM_REPORTVARS is zero), then these macro variables are not defined.	SAS name
HPF_TIMEID	Time ID variable name.	SAS name
HPF_TIMEID_FORMAT	SAS format of the time ID variable.	SAS format name

Name	Description	Format
HPF_SEASONALITY	Integer length of the seasonal cycle A seasonality of 1 implies no seasonality.	Positive integer
HPF_INTERVAL	Interval of the time ID variable (for example, MONTH).	SAS time interval
HPF_DATASTART	Start date, datetime, or time value of the project The starting time ID value of the project input data set (&HPF_LIBNAME.&HPF_DATASET).	SAS date, datetime, or time value
HPF_DATAEND	End date, datetime, or time value of the project The ending time ID value of the project input data set (&HPF_LIBNAME..&HPF_DATASET).	SAS date, datetime, or time value
HPF_SETMISSING	Controls the interpretation of missing values.	
HPF_TRIMMISS	Controls whether missing values are trimmed.	
HPF_ZEROMISS	Controls the interpretation of zero values.	
HPF_NUM_LEVELS	Number of levels in the hierarchy The levels of the hierarchy are numbered from 1 (the top of the hierarchy) to &HPF_NUM_LEVELS (the leaves of the hierarchy). If there is no hierarchy, then the number of levels is 1.	Positive integer
HPF_LEVEL_BYVARS#	Space-delimited list of BY variable names associated with the n^{th} level, where *n* ranges from 1 to &HPF_NUM_LEVELS. The variables names are separated by a single space. The BY variables at level 1 (the top) are stored in HPF_LEVEL_BYVARS1. The BY variables for the lowest level (the leaves) are stored in HPF_LEVEL_BYVARS&HPF_NUM_LEVELS.	SAS name

Name	Description	Format
HPF_LEVEL_DATAWHERE#	Input data filtering WHERE clause for the nth level, where *n* ranges from 1 to &HPF_NUM_LEVELS These WHERE clauses can be used to subset the input data sets for each level in the hierarchy to obtain information about the currently selected node. The WHERE clause at level 1 (the top) is stored in HPF_LEVEL_DATAWHERE1. The WHERE clause at the lowest level (the leaves) is stored in HPF_LEVEL_DATAWHERE&HPF_NUM_LEVELS. *Note:* You must unquote this macro variable. For example, `%unquote (&&HPF_CURRENT_DATAWHERE&n);`	SAS WHERE clause
HPF_LEVEL_OUTWHERE#	Output data filtering WHERE clause for the nth level where *n* ranges from 1 to &HPF_NUM_LEVELS These WHERE clauses can be used to subset the input data sets for each level in the hierarchy to obtain information about the currently selected node. The WHERE clause at level 1 (the top) is stored in HPF_LEVEL_OUTWHERE1. The WHERE clause at the lowest level (the leaves) is stored in HPF_LEVEL_OUTWHERE&HPF_NUM_LEVELS. *Note:* You must unquote this macro variable. For example, `%unquote (&&HPF_LEVEL_OUTWHERE&n);`	SAS WHERE clause
HPF_LEVEL_LIBNAME#	SAS library reference for the nth level, where *n* ranges from 1 to &HPF_NUM_LEVELS The library reference at level 1 (the top) is stored in HPF_LEVEL_LIBNAME1. The library reference at the lowest level (the leaves) is stored in HPF_LEVEL_LIBNAME&HPF_NUM_LEVELS.	SAS LIBNAME

Name	Description	Format
HPF_LEVEL_NSERIES#	Number of series associated with the nth level, where *n* ranges from 1 to &HPF_NUM_LEVELS The number of series at level 1 (the top) is stored in HPF_LEVEL_NSERIES1. The number of series at the lowest level (the leaves) is stored in HPF_LEVEL_NSERIES&HPF_NUM_LEVELS	Positive integer

Diagnose Macro Variables

The following variables describe the current diagnose settings for the active project.

Table 10.7 Diagnose Macro Variables

Name	Description	Format
HPF_DIAGNOSE_ INTERMITTENT	Intermittency threshold values for diagnose tests.	Positive number
HPF_DIAGNOSE_ SEASONTEST	Seasonality significance level for diagnose tests.	*p*-value

Model Selection Macro Variables

The following variables describe the current model selection settings for the active project.

Table 10.8 Model Selection Macro Variables

Name	Description	Format
HPF_SELECT_CRITERION	Statistic of fit to use for model selection.	
HPF_SELECT_HOLDOUT	Absolute number of observations to holdout for selection. Zero implies that the model fit is used for selection.	
HPF_SELECT_ HOLDOUTPCT	Percentage of observations to hold out for selection.	Positive integer
HPF_SELECT_MINOBS_ NON_MEAN	Minimum number of observations to require for a non-mean model.	Positive integer
HPF_SELECT_MINOBS_ TREND	Minimum number of observations to require for a trend model.	Positive integer

Forecast Macro Variables

The following variables describe the current forecasting settings for the active project.

Table 10.9 Forecast Macro Variables

Name	Description	Format
HPF_LEAD	Integer number of time periods to forecast the length of the forecast horizon or lead.	Positive integer
HPF_FORECAST_ALPHA	Confidence level size.	*p*-value
HPF_BACK	Integer number of time periods to ignore when forecasting.	Positive integer

Reconciliation Macro Variables

The following variables describe the current reconciliation settings for the active project.

Table 10.10 Reconciliation Macro Variables

Name	Description	Format
HPF_RECONCILE_LEVEL	Index of the reconciliation level. The reconciliation level index ranges from 1 to &HPF_NUM_LEVELS, depending on the level of reconciliation.	Positive integer
HPF_RECONCILE_BYVAR	BY variable associated with the level used for reconciliation.	SAS name
HPF_RECONCILE_METHOD	Reconciliation method	SAS name
HPF_LEVEL_RECONCILE_DATASET#	The reconciled forecast data set for each level.	
HPF_LEVEL_RECONCILE_STATISTICS#	The reconciled statistics data set for each level.	
HPF_LEVEL_RECONCILE_SUMMARY#	The reconciled summary data set for each level.	

Active Series Macro Variables

The active series macro variables differ from the others in that they depend on the node of the tree (in the forecasting hierarchy) that is currently selected by the user. Because of this dependency, you cannot use these macro variables outside of SAS Forecast Server.

Table 10.11 *Active Series Macro Variables*

Name	Description	Format
HPF_CURRENT_LEVEL	The level index number associated with the current level. The current level number ranges from 1 to &HPF_NUM_LEVELS, depending on the currently selected level of the hierarchy.	Positive integer
HPF_CURRENT_LIBNAME	The SAS library reference associated with the currently selected level of the hierarchy.	SAS LIBNAME
HPF_CURRENT_LEVEL_START	Start date, datetime, or time value of the current level. The starting time ID value of the input data set for the currently selected level of the hierarchy.	SAS date, datetime, or time value
HPF_CURRENT_LEVEL_END	End date, datetime, or time value of the current level. The ending time ID value of the input data set for the currently selected level of the hierarchy.	SAS date, datetime, or time value
HPF_CURRENT_LEVEL_NSERIES	Number of series (or nodes) associated with the currently selected level of the hierarchy.	Positive integer
HPF_CURRENT_SERIESSTART	Start date, datetime, or time value of the current node. The starting time ID value of the series for the currently selected node of the hierarchy.	SAS date, datetime, or time value
HPF_CURRENT_SERIESEND	End date, datetime, or time value of the current node. The ending time ID value of the series for the currently selected node of the hierarchy.	SAS date, datetime, time value

Name	Description	Format
HPF_CURRENT_DATAWHERE	Input data WHERE clause for the currently selected node. This WHERE clause can be used to subset the input data set to obtain information about the currently selected node of the hierarchy. *Note:* You must unquote this macro variable. For example, `%unquote (&HPF_CURRENT_DATAWHERE);`	SAS WHERE clause
HPF_CURRENT_OUTWHERE	Output data WHERE clause for the currently selected node. This WHERE clause can be used to subset the output data sets to obtain information about the currently selected node. *Note:* You must unquote this macro variable. For example, `%unquote (&HPF_CURRENT_OUTWHERE);`	SAS WHERE clause
HPF_NUM_CURRENT_BYVARS	Number of BY variable names for the currently selected level of the hierarchy.	Positive integer
HPF_CURRENT_BYVARS	Space-delimited list of BY variable names for the currently selected level of the hierarchy. The macro variable is always defined; but if there are no BY variables, HPF_CURRENT_BYVARS is set to NULL.	List of SAS names separated by a single space
HPF_CURRENT_BYVARS#	Individual BY variable names for the current level.	SAS name
HPF_CURRENT_DEPVAR	Dependent variable name associated with the currently selected node of the hierarchy. This variable is contained in the list of dependent variables (HPF_DEPVARS).	SAS name
HPF_CURRENT_HORIZON	Horizon date, datetime, or time value of the current node The time ID value of the start of the multi-step ahead forecast for the currently selected node of the hierarchy.	SAS date, datetime, or time value

Sample Reports to Get You Started

Note: You must deploy the sample reports before they are available for use in SAS Forecast Studio. For more information, see "Using the Sample Reports" on page 79.

To understand how to use dynamic parameters, multi-value parameters, and manually assigned metadata libraries, see these categories of sample reports in the Reports and Stored Processes dialog box in SAS Forecast Studio.

Using Dynamic Parameters
: This example shows all of the special parameter names that you can use with dynamic lists and demonstrates how to configure these parameters.

Using Libraries
: This example shows how libraries that are not pre-assigned can be used in reports.

Using Package Files
: This example shows how to add files to the generated results package. These files can be accessed by saving the report output to your local file system.

Working with List Variables
: This example shows how list variables are passed to programs. In addition, it demonstrates how to reformat these variables automatically for compatibility with the format used by the previous release.

Initializing a Report

When you create reports, the header portion of the report's source code file must contain macro calls to initialize the SAS Forecast Server macros and ODS output as follows:

```
*ProcessBody;

/*------------------------------------------------------------*/
/*- initialize the HPF stored process support                -*/
/*------------------------------------------------------------*/
%hpfstp();

/*------------------------------------------------------------*/
/*- initialize the ODS output                                -*/
/*------------------------------------------------------------*/
%HPF_InitODSOutput();

%stpbegin;
```

Chapter 11

Miscellaneous Administration Tasks

Cloning a Forecasting Environment

When you clone a forecasting environment, you copy the structure and contents of an existing environment. The following instructions do not cover issues that can arise when you copy across files systems or different operating environments. Instead, these instructions assume that you will perform any necessary migrations of your file formats.

Copy an Environment Definition

1. In the file system location where you plan to save the new environment, create any necessary directories and set the appropriate permissions on each directory.
2. From the existing environment directory, recursively copy the **`Libraries`** and **`Config`** subdirectories (if they exist) to the directory for the new environment.
3. If necessary, update any directory paths in the configuration details in the **`Config`** subdirectory.
4. Register the location of the new environment in either of the following ways:

- Use the Register action in the SAS Forecast Server Plug-in for SAS Management Console. For more information, see the Help for the SAS Forecast Server plug-in.
- Use the FSREGENV macro. For more information, see "FSREGENV Macro" on page 149.

CAUTION:

Do not use the Create action to register an environment. The Register action updates the metadata of the environment without changing the content in the file system. The Create action creates new content.

5. (Optional) Configure the Reports tree for the new environment to match the configuration of the existing environment.

   Note: How you configure the reports tree and whether you can simply share the same tree with the existing environment depends on the following considerations:

   - the types of stored processes that you intend to use
   - whether the two environments are on the same server

   For more information, see "Report Administration Tasks" on page 75.

6. In the metadata view of SAS Folders, compare the permissions on the existing and new environments and their parent folders. If necessary, update the permissions for the new environment and its parent folder.

   Note: The metadata for environments is saved in **`/System/Application/SAS Forecast Server/Forecast Server 4.1/Environments`**.

Copy the Existing Environment's Content

You can copy project content from one environment to another in the following ways:

- back up and restore projects using archives
- copy projects using the tools in the SAS Forecast Server plug-in for SAS Management Console

You can perform these tasks using either the SAS Forecast Server macros or the SAS Forecast Server Plug-in for SAS Management Console. For more information about the available macros, see "The SAS Forecast Server Macros" on page 108.

Using the SAS Forecast Server Plug-in for SAS Management Console, you can archive a project, unarchive a project, and copy a project. You can also perform these actions in batch. For example, to copy multiple projects from the existing environment to the new environment, select the batch copy action and then select all of the projects in the existing environment. When prompted, select the new environment.

Note: Both tasks (backing up and restoring content with an archive or copying projects) result in creating new metadata for the cloned environment.

Using Custom Time Intervals

Create the Custom Time Interval

Although SAS Forecast Server includes a variety of time intervals, your site might use custom time intervals.

To create a custom interval:

1. Using a DATA step program or a SAS editor, create the data set that defines the custom interval. The data set must include the following information:

   - a BEGIN variable that specifies the date or datetime at which each period begins. If raw data contains date values (the number of days since January 1, 1960), use these date values for the BEGIN variable and assign a date format. If the raw data contains datetime values (the number of seconds since midnight January 1, 1960), use datetime values for the BEGIN variable and assign a datetime format.

     You can also define the end of each period (by using an END variable) and the seasonal cycle (by using a SEASON variable). The END variable must use the same format as the BEGIN variable. If you do not specify an END variable, then the implied value of END for each observation is one less than the value of the BEGIN variable at the next observation.

   - The span of the custom interval data set must include any dates or times that are necessary for performing calculations on the time series, such as forecasting and any other operations that might extend beyond the series (such as filters).

   CAUTION:
   Errors will occur if the date or datetime values (in the actual data, in forecasts, or in plots) are outside the range of the custom interval definition. For your custom interval, the values for the BEGIN variable must start with the earliest historical data and extend into the future for at least three forecasting horizons. For example, if the historical data includes all of the business days for 2011 and the forecasts are for the first 12 business days in 2012, the values of the BEGIN variable must range from the first business day of 2011 through the 36th business day of 2012.

2. To define the name and location of your custom interval, specify the INTERVALDS= system option in either of these files:

   - in the config file (**`sasv9_usermods.cfg`** or **`sasv9.cfg`**) that is used by the SAS Workspace Server.

   - in the autoexec file (for example, **`appserver_autoexec_usermods.sas`**) that is used by the SAS Workspace Server.

   In SAS Forecast Server, the syntax for the INTERVALDS= system option is more restrictive than the syntax provided in the *SAS System Options: Reference*. Here is the syntax:

   INTERVALDS=(*interval-1=libref.dataset-name-1<interval-n=libref.dataset-name-n>*)

   The name for the *interval* must be 1–32 characters. The name cannot contain underscores or embedded numeric characters, except that the name can end with one or more digits. For example, StoreHours12 is a valid name, and Store12Hours is an

invalid name. When you specify multiple intervals, each interval name must be unique. The value of the interval is the data set that is named in *libref.dataset-name*.

libref.dataset-name specifies the libref and data set name of the file that contains the custom interval.

In this example, the StoreHours interval is associated with the StoreHoursDS data set.

```
options intervalds=(StoreHours=CustIntLib.StoreHoursDS);
```

After completing these steps, the custom interval should appear in the list of available time intervals the next time you start SA Forecast Studio. For example, you can select a time interval when creating a new project in the New Project wizard. If a custom interval is not available, check the SAS log for errors in the definition of the custom interval.

Example 1: StoreHours Custom Interval

This example shows how to set up a custom interval for data that is recorded hourly during the hours of 9AM to 6PM Monday through Friday and 9AM to 1PM on Saturday.

1. Log on to the SAS Workspace Server. Use a text editor to add the following code to the **`C:\SAS\Config\Lev1\SASApp\sasv9_usermods.cfg`** file:

   ```
   -intervalds (StoreHours=CustIntLib.StoreHoursDS);
   ```

2. Start a SAS session on the workspace server. In this SAS session, complete these steps:

   a. Assign the LIBNAME CustIntLib to the directory where the data set for the custom interval (in this example, StoreHoursDS) is stored.

   b. Submit the following code to create the data set:

   ```
   data CustIntLib.StoreHoursDS(keep=BEGIN END); 1
      start = '01JAN2009'D;
      stop  = '31DEC2009'D;
      do date = start to stop; 2
         dow = WEEKDAY(date);
         datetime=dhms(date,0,0,0);
         if dow not in (1,7) then
            do hour = 9 to 17;
               begin=intnx('hour',datetime,hour,'b');
               end=intnx('hour',datetime,hour,'e');
               output;
            end;
         else if dow = 7 then
            do hour = 9 to 13;
               begin=intnx('hour',datetime,hour,'b');
               end=intnx('hour',datetime,hour,'e');
               output;
            end;
      end;
      format BEGIN END DATETIME.; 3
   run;
   ```

   a In the DATA statement, the name StoreHoursDS is specified for the data set. The KEEP= option specifies that only the BEGIN and END variables should be included in the data set.

The START= and STOP= options specify the date of the first and last observations in the data set.

b The DO loop analyzes each observation from 01JAN2009 through 31DEC2009 (inclusive).

- Use the WEEKDAY function to determine the day of the week for a specific date. By default, the days of the week are specified as 1 for Sunday, 2 for Monday, and so on. You need this information later in the DO loop to assign the correct store hours to the day of the week. For example, if the day of the week is not 1 (Sunday) or 7 (Saturday), then the store hours are 9 to 17. If the day of the week is 7 (Saturday), then the store hours are 9 to 13.
- Use the DHMS function to return a SAS datetime value for the date.
- Use the INTNX function to return the datetime value of the beginning of the interval that is *n* intervals from the interval that contains the given datetime value.

c Finally, the FORMAT statement specifies a DATETIME. format for the values of the BEGIN and END variables.

Example 2: Creating a Custom Interval Definition Interactively

To create a custom interval definition interactively:

1. Start a SAS session on the SAS Workspace Server.
2. Click the New Library icon in the toolbar. The New Library dialog box appears.
3. Type the LIBNAME that you used when you specified the INTERVALDS system option.
4. Select the path where the custom interval will be stored. Click **OK** to close the Select and New Library dialog boxes.
5. In the SAS Explorer, right-click the library that you just created and click **New**. The New Member dialog box appears.
6. Select **Table** and click **OK**. An empty table opens in Viewtable.
7. Right-click the **A** at the type of column A and select **Column Properties**.
8. Specify these properties:
   - Change the name from **A** to **BEGIN**.
   - Select **Numeric** as the type.
   - Specify **DATE9.** as the format and informat.

   Click **Close**.
9. Click the first cell in the BEGIN column to edit that cell. Type **01JAN2011** and press ENTER.
10. In the second cell, enter the date of the next period. Continue entering dates as needed. Then close the Viewtable window. When prompted, click **Yes** to save your changes. In the Save As window, select the library that you assigned in step 3.
11. For the member name, type the name of the data set that you specified in the INTERVALDS system option.
12. Click **Save** to save the data set.

Add a Custom Format to SAS Forecast Server

Although SAS Forecast Server includes a variety of formats, you might have additional formats available at your site. Using the FORMAT procedure, you create custom formats and then make these formats available in SAS Forecast Studio. For more information about this procedure, see *Base SAS Procedures Guide*.

To create a customized format:

1. In PROC FORMAT, specify where to store the format by using the LIBRARY= option.
   - To store the format in a library or catalog in the default location, write the following SAS code:

     ```
     libname library
     config-dir/Levn/SASApp/SASEnvironment/SASFormats;
     proc format library=library;
     ...
     ```

   - To store the user-defined formats in the **C:\myfmts** location on the SAS server, write the following SAS code:

     ```
     libname library "c:\myfmts";
     proc format library=library;
     ...
     ```

     By default, the catalog name for this library is **Formats** (unless you changed the default value to a different catalog name).
2. Use the VALUE statement to, define the format. For example, the following code defines the $LINE format. The special code **Line1**, **Line2**, and so on, are converted to the corresponding product line.

   ```
   value $ line
   'Line1'='Product Line1'
   'Line2'='Product Line2'
   'Line3'='Product Line3'
   'Line4'='Product Line4'
   'Line5'='Product Line5';
   ```

3. Define the format library for the SAS Workspace Server that is used by SAS Forecast Server.
   - If the customized format is stored in the default library (***SAS-configuration-directory*/Lev1/SASApp/SASEnvironment/SASFormats)**, then no further modifications are required.
   - If the customized format is in a different location, you must configure SAS to search the **C:\myfmts** library in addition to the default library.

To search a different format library:

1. Open the configuration file in the following default location:

Windows

***SAS-configuration-directory*\Levn\SASApp\sasv9_usermods.cfg**

UNIX

***SAS-configuration-directory*/Levn/SASApp/sasv9_usermods.cfg**

2. In the sasv9_usermods.cfg file, use the SET= system option to define the library definition. In the following code, the SET= option associates the FSFMTS catalog with the **C:\myfmts** directory. Then set the FMTSEARCH= system option to FSFMTS, so that SAS Forecast Server searches this format catalog.

```
-set FSFMTS ("c:\myfmts")
-fmtsearch (FSFMTS)
```

   When you restart the SAS Workspace Server, the system resolves any references to custom formats that are stored in **C:\myfmts**.

Adding Start-Up and Shutdown Code for an Environment

Create the Environment Code

You might want to run specific SAS code every time you start up or shutdown a particular environment.

To create this code:

1. Open the SAS Program Editor and write the SAS code that you want to use.
2. Save this code to either the startup.sas or shutdown.sas file in the ***environment-directory*\Config** directory.

Note: As with all server code, update this code when no one at your site is using the specified environment. A good time to make these code updates is during a maintenance period.

How to Configure This Code to Run

Site administrators can specify whether this code should run for a SAS deployment. If the code will not be run, a message stating that code execution is not permitted appears. Due to security considerations, this code is ignored by default.

To enable the use of this code:

1. Start SAS Management Console and log on as a SAS administrator (for example, sasadm@saspw).
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.

4. Click the **Settings** tab.
5. In the selection pane, select **Forecast Server**.
6. From the **Environment scripts** drop-down list, select **Execute**.
7. Restart SAS Forecast Server for these changes to take effect.

Adding Start-Up and Shutdown Code to a Project

About This Code

You can customize a project by specifying SAS code to run when the project is opened or closed. For example, you specify the MPRINT system option to run when the project opens. Now, any SAS statements that are generated when the user runs the macros for SAS Forecast Server are traced for debugging. When you close a project, you might want SAS Forecast Studio to remove any temporary files that were created, make backups of files, or automatically export your results to an external data set.

The code is saved to the startup.sas and shutdown.sas files in the ***project-directory*\Config** directory. The code can be modified only when the project is open in SAS Forecast Studio.

Note: In SAS Management Console, this functionality is referred to as project scripts.

How to Configure This Code to Run

Site administrators can specify whether this code should run for a SAS deployment. If the code will not be run, a message stating that code execution is not permitted appears. Due to security considerations, this code is ignored by default.

To run this code:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, select **Forecast Server**.
6. From the **Project scripts** drop-down list, select **Execute**.
7. Restart SAS Forecast Server for these changes to take effect.

Remove a Configuration with the SAS Deployment Manager

To remove a configuration:

1. Navigate to the SAS_HOME directory. For example, on a Windows system, this is the `C:\Program Files\SAS\SASDeploymentManager\9.2` folder.
2. Double-click sasadm.exe to launch the SAS Deployment Manager.
3. Select **Remove existing configuration**, and then click **Next**.
4. Select a **Configuration directory**, and then click **Next**.
5. Specify your connection information to the SAS Metadata Server, and then click **Next**.
6. Select the product that you want to unconfigure, and then click **Next**. For example, to unconfigure SAS Forecast Server, select **Forecast Server 4.1**.
7. (Optional) The SAS Forecast Server stores metadata describing user-defined environments and projects in the SAS Metadata Server. If you are removing a configuration for SAS Forecast Server, you can also select to unregister the user content. This option removes the SAS Forecast Server metadata when you remove the configuration. In either case, the project content is not deleted as you remove the configuration, and you can use it to recreate the removed metadata later if desired.

   Note: You can use the SAS Forecast Server Plug-in for SAS Management Console to register and unregister user content. For more information about registering and unregistering content, see the Help for the SAS Forecast Server Plug-in for SAS Management Console.
8. In the **Summary** screen, click **Start**.
9. Restart SAS Forecast Server for these changes to take effect.

Chapter 12

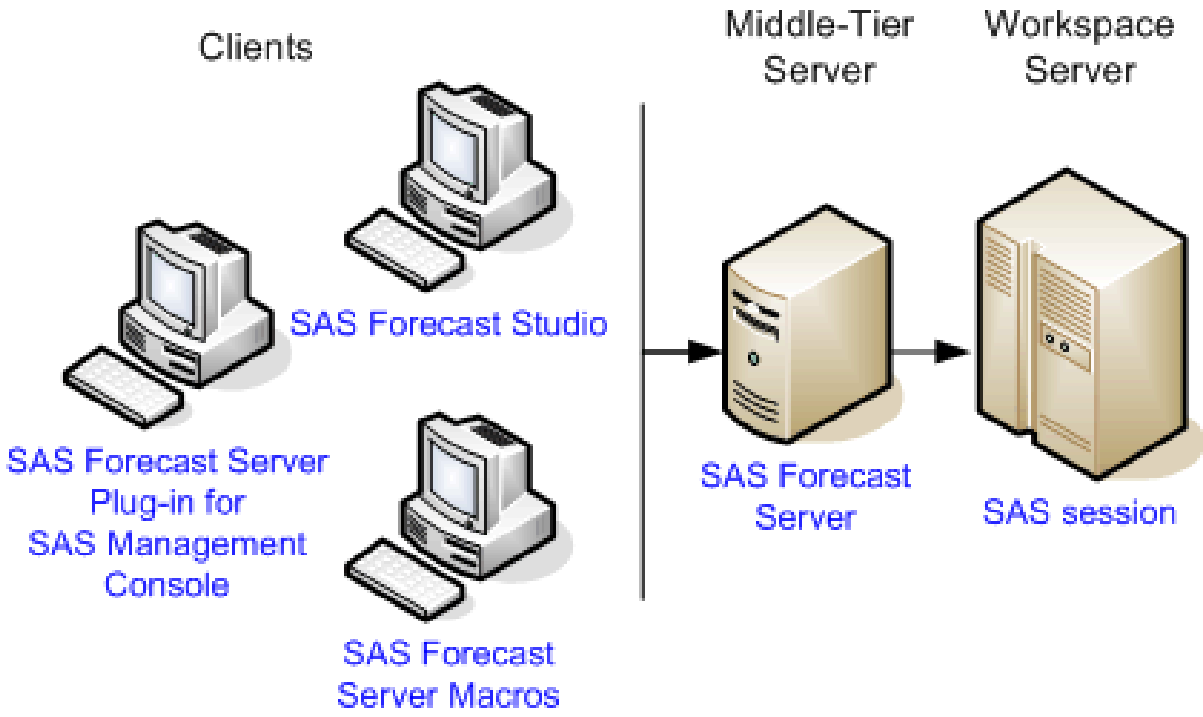
The SAS Forecast Server Administrative Macros

The SAS Forecast Server Macros

Summary of the SAS Forecast Server Macros

The SAS Forecast Server macros provide a SAS language interface to SAS Forecast Server. (This is similar to the graphical user interface that SAS Forecast Studio provides to SAS Forecast Server.) Effectively, the macros are an alternative client to both SAS Forecast Studio and the SAS Forecast Server Plug-in for SAS Management Console. Some macros enable you to create projects with features identical to a project that you created in SAS Forecast Studio. However, most of the macros focus on the management of environments and projects.

The following figure shows a schematic representation of how the macros interact with the other components of SAS Forecast Server.



You can run these macros on any system that has an installation of SAS Foundation. On Windows, the macros reside in an autocall library in **`!SASROOT/forecastbat/sasmacro`**. On UNIX, they reside in the **`SASFoundation/9.3/sasautos`** directory.

The SAS Forecast Server macros have the same requirements for the SAS environment as any other client. These macros must have access to a sas-environment.xml file that defines the list of SAS environments, or you will get an error. For all macros, the sasEnvironment argument is used to identify the SAS environment. The value for this argument is the symbolic name of the SAS environment. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file. A symbolic name is used because the name is independent of your locale.

The following table lists the SAS Forecast Server macros.

Macro Name	Description
FSADDEVT	Defines an event and adds it to the event repository for the specified project.
FSCLEAR	Clears project information currently stored in global macro variables.
FSCOPY	Copies a SAS Forecast Server project to a new destination.
FSCREATE	Creates a new SAS Forecast Server project in batch mode.
FSDELARC	Deletes an archived SAS Forecast Server project.
FSDELENV	Deletes an existing SAS Forecast Server environment.
FSDELEVT	Deletes specified events from the event repository.
FSEXPALL	Exports all SAS Forecast Server projects to archive files.
FSDELPRJ	Deletes an existing SAS Forecast Server project.
FSEXPORT	Exports a single SAS Forecast Server project to an archive file.
FSGETENV	Retrieves the metadata about the SAS Forecast Server environments.
FSGETPRJ	Retrieves the metadata about the SAS Forecast Server projects.
FSGETURP	Creates a log file that lists the names of any unregistered projects in a specified environment.
FSIMPALL	Imports all SAS Forecast Server projects listed in a data set from archived files.
FSIMPORT	Imports a SAS Forecast Server project from an archived file.
FSLOAD	Opens an existing SAS Forecast Server project, and loads global macro variables that describe the project.

Macro Name	Description
FSMIGALL	Migrates all existing SAS Forecast Server projects to the current version of SAS Forecast Server.
FSMIGPRJ	Migrates an existing SAS Forecast Server project to the current version of SAS Forecast Server.
FSMOVE	Moves a SAS Forecast Server project to a new destination.
FSNEWENV	Creates a new SAS Forecast Server environment.
FSREGENV	Registers an existing directory structure as an environment, optionally registering all projects found within.
FSREGPRJ	Registers a project in metadata.
FSREN	Renames a single SAS Forecast Server project.
FSRUNPRJ	Opens an existing SAS Forecast Server project, and runs the project at a given stage.
FSSETOWN	Assign the owner of a project.
FSSETPUB	Determines whether public access to a SAS Forecast Server project should be enabled.
FSUNREG	Unregisters an existing SAS Forecast Server project from the metadata server.
FSUNRENV	Unregisters an environment.

Note: The SAS macros are automatically installed with SAS Forecast Server. For example, on Windows, the default installation location is the **`C:\Program Files\SAS\SASFoundation\9.3\forecastbat\sasmacro`** directory.

Encrypting Your Passwords

It is not recommended that you use plain text passwords in the code for your macro variables. Instead, use PROC PWENCODE to obtain an encrypted password.

For example, to encrypt the password “SASmeta1,” submit the following code in the SAS Program Editor:

```
proc pwencode in="SASmeta1";
run;
```

Copy the encrypted password from the SAS log and paste this password into your macro code.

For more information about the PWENCODE procedure, see the *Base SAS Procedures Guide*.

Dictionary

FSADDEVT Macro

The FSADDEVT macro defines an event and adds it to the event repository for the specified project. All types of events and their attributes can be defined.

Syntax

%FSADDEVT(PROJECTNAME=,USER=,PASSWORD=,EVENTNAME=,EVENTTYPE, [,options]);

Details

Required Arguments

The following arguments must be used with the FSADDEVT macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

EVENTNAME= *event-name*
specifies the name of the event to be created.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user's password that you use to log on to SAS Forecast Studio.

EVENTTYPE= POINT | RAMP | LEVELSHIFT | TEMPORARYCHANGE | COMBINED
specifies the type of event. For more information about each type of event, see the *SAS Forecast Studio: User's Guide*.

Note: POINT events are called pulse events in SAS Forecast Studio.

Options

The following options can be used with the FSADDEVT macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The value for this argument is case sensitive. The default environment is **`Default`**.

EVENTCHANGEPARAMETER = *value*
specifies the change for a temporary change event. This option does not have any effect on other event types. Valid values range from 0 to 1 (inclusive). The default value is 0.5.

EVENTCOMBINATION=*event-name1 event-name2*
specifies the events to include in the combined event. This option is valid only when the event type is COMBINED. Event names are delimited by a space.

EVENTCOMBINATIONRULE= ADD | MAX | MIN | MINNZ | MINMAG | MULT
specifies which combination rule to use when the defined event has several values that overlap in the same time period. Here is a description of the valid values:

ADD	add the overlapping values
MAX	use the maximum value
MINNZ	use the minimum nonzero value
MINMAG	use the value with the least magnitude
MULT	multiply the overlapping values

EVENTDESCRIPTION=*text*
specifies a description of the event.

EVENTDIFF = YES | TRUE | NO | FALSE
specifies whether to use the same differencing value that is used in the model. This option applies only to ARIMA models. The default value is **YES**.

EVENTOCCURRENCESAFTER = *value*
specifies the duration of the component that occurs after the event. For example, EVENTOCCURRENCESAFTER=4 specifies that the event ends four periods after the time that you specified for the event. If you specified more than one date or time for the event, the EVENTOCCURRENCESAFTER option applies to all of them.

EVENTOCCURRENCESBEFORE = *value*
specifies the duration of the component that occurs before the event. For example, EVENTOCCURRENCESBEFORE=4 specifies that the event starts four periods before the date and time that you specified for the event. If you specified more than one date or time for the event, the EVENTOCCURRENCESBEFORE option applies to all of them.

EVENTOCCURRENCESAFTERGROWTH = YES | TRUE | NO | FALSE
specifies the slope for the component that occurs after the event. This option is valid only if the event type is RAMP or TEMPORARYCHANGE. The YES | TRUE values indicate growth, and the NO | FALSE values indicate decay.

EVENTOCCURRENCESBEFOREGROWTH = YES | TRUE | NO | FALSE
specifies the slope for the component that occurs before the event. This option is valid only if the event type is RAMP or TEMPORARYCHANGE. The YES | TRUE values indicate growth, and the NO | FALSE values indicate decay.

EVENTPERIODINTERVAL = *time-interval*
specifies the time interval by which the event repeats infinitely.

EVENTPERIODICFROMSTART = *date-time-value*
specifies the starting point for the recurrence. The interval is specified by the EVENTPERIODINTERVAL option.

EVENTPERIODICFROMSTARTINTERVAL = *time-interval*
specifies the time interval for an event that repeats infinitely. The starting point is specified by the EVENTPERIODICFROMSTART option.

EVENTPERIODICITYEND = *date-time-value*
specifies the end date or time for the periodic interval. When you specify the EVENTPERIODICITYSTART, EVENTRANGEINTERVAL, and EVENTPERIODICITYEND options, then the periodicity for the recurrence is finite.

EVENTPERIODICITYOCCURENCES = *integer-value*
specifies the number of times that the event recurs. The recurrence starts at the date or time specified by the value of the EVENTPERIODICITYSTART option and occurs at the interval specified by the value of the EVENTRANGEINTERVAL option. If you specify this option and the EVENTPERIODICITYEND option, the value of the EVENTPERIODICITYEND option is ignored.

EVENTPERIODICITYSTART = *date-time-value*
specifies the start date or time for the periodic interval. When you specify the EVENTPERIODICITYSTART, EVENTRANGEINTERVAL, and EVENTPERIODICITYEND options, then the periodicity for the recurrence is finite.

EVENTRANGEINTERVAL = *time-interval*
specifies the time interval at which the event repeats. The start and end dates are specified by the EVENTPERIODICITYSTART and EVENTPERIODICITYEND options.

EVENTREQUIRED = NO | MAYBE | YES | UNDEF
specifies whether the event should be used when SAS Forecast Server automatically fits a model. The default value is **`NO`**. Here are the descriptions of the valid values:

NO	specifies that the events be included in the model as long as the event parameters are significant and the increment of the value of criterion exceeds the specified threshold.
MAYBE	specifies that the events be included in the model as long as the event parameters are significant.
YES	specifies that the events be included in the model as long as the model can be diagnosed.
UNDEF	specifies that the events not be included in the model.

EVENTTIMINGSHIFT = *integer-value*
specifies how long to delay the onset of the event. This delay is calculated from the start date or time of the event. Negative values can be used to specify an onset before the start date or time.

EVENTTIMINGVALUES = *value1 value2 . . .*
specifies the dates and times of the event. Values must be separated by a space. Here are the valid values:

- Dates in the ddmmmyyyy format (for example, 01JAN2012)
- Date-time values in the ddmmmyyyy:hh:mm:ss format (for example, 01JAN2012:04:21:15)
- Event keywords, including holiday names and seasonal events. For a list of these keywords, see the "Event Definitions" topic in the HPFEVENTS procedure chapter in the *SAS High-Performance Forecasting: User's Guide*.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSADDEVT global macro variable indicates whether a new event was created or whether it failed due to errors.

Example

```
%fsaddevt(projectname=Project1,
          user=sasuser,
          password=saspass,
          environment=Default,
          eventname=evt2,
          eventtype=ramp,
          eventdescription=sample event,
          eventTimingValues=01MAR1955,
          eventRequired=YES
          );
```

FSCLEAR Macro

The FSCLEAR macro clears project information currently stored in global macro variables. Use the FSLOAD macro to add project information to a global macro variable.

Syntax

%FSCLEAR();

Details

Required Arguments

There are no arguments required for this macro.

Options

This macro does not use any optional parameters.

Example

```
%fsclear();
```

FSCOPY Macro

The FSCOPY macro copies a SAS Forecast Server project to a new destination. Both servers must run the same version or a later version of SAS Forecast Server.

Syntax

%FSCOPY (SOURCEPROJECTNAME=, DESTINATIONPROJECTNAME=, USER=, PASSWORD=, REMOTEARCHIVEFOLDER= SOURCEENVIRONMENT=, DESTINATIONENVIRONMENT= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSCOPY macro. The required arguments are separated by commas.

SOURCEPROJECTNAME= *source-project-name*
specifies the name of the SAS Forecast Server project on the source host. The name must be a valid SAS name.

DESTINATIONPROJECTNAME= *destination-project-name*
specifies the name of the SAS Forecast Server project on the destination host. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

REMOTEARCHIVEFOLDER= *directory-name*
specifies the directory where the archived projects of the source SASWorkspace Server can be found on the destination SAS Workspace Server (for example, **\ \\sourceserver\SAS\ ForecastStudio\Archives**).

Note: The user must have Read and Write privileges to the REMOTEARCHIVEFOLDER= directory.

SOURCEENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment to be used as source. This option is required only if there are multiple environments available to the user.

DESTINATIONENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment to be used as destination. This option is required only if there are multiple environments available to the user.

Options

The following options can be used with the FSCOPY macro. Options must follow the required arguments and are separated by commas.

ARCHIVEFOLDER= *directory-name*
specifies the directory where to save the archive. If not specified, the default location is used.

CPORT= YES | TRUE | NO | FALSE
exports data sets and catalogs using CPORT. You must specify this option if the source and destination environments are running on different operating systems. The default is CPORT=NO.

NOWARN= YES | NO
suppresses errors if the SAS Forecast Server project does not exist. Possible values are YES | NO. The default is NOWARN=NO.

PRINTVERSIONONLY= 0 | 1
When PRINTVERSIONONLY=1, it prints the version to the log and exits.

TMPARCHIVENAME= *project-name*
specifies the name of the temporary archived project. The default is _fs_tmp_archive. The name must not match an existing archive.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSCOPY global macro variable indicates whether the FSCOPY macro terminates successfully or encounters errors: &FSCOPY= SUCCESS | ERROR.

Example

```
%fscopy(sourceprojectname=prdat1,
destinationprojectname=prdat2,
user=sasuser,
password=saspass,
remoteArchiveFolder=\\sourceserver\SAS\ForecastStudio\Archives,
sourceEnvironment=Default,
destinationEnvironment=Default,
);
```

FSCREATE Macro

The FSCREATE macro creates a new SAS Forecast Server project in batch mode.

Syntax

%FSCREATE (PROJECTNAME=, USER=, PASSWORD=, DATA=, ID=, VAR= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSCREATE macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

DATA= *SAS-data-set*
specifies the name of the input SAS data set. The value that you enter for this argument is not case sensitive.

ID=*variable*
specifies the time ID variable

VAR= *variable*
specifies one or more dependent variables. Multiple variable names can be specified only if HIERARCHY= NO | FALSE. Multiple variable names are separated by a space.

Options

The following options can be used with the FSCREATE macro. Options must follow the required arguments, and are separated by commas.

ACCUMULATE= TYPE [TYPE(*var1var2)* TYPE(*var3var4*) ...]
specifies the accumulation options for the dependent, input, and reporting variables.

Here are the valid values for TYPE:

NONE	specifies that no accumulation occurs; the ID variable values must be equally spaced with respect to the frequency.
TOTAL	specifies that observations are accumulated based on the total sum of their values. This is the default value.
AVERAGE | AVG	specifies that observations are accumulated based on the average of their values.
MINIMUM | MIN	specifies that observations are accumulated based on the minimum of their values.
MEDIAN | MED	specifies that observations are accumulated based on the median of their values.
MAXIMUM | MAX	specifies that observations are accumulated based on the maximum of their values.
N	specifies that observations are accumulated based on the number of nonmissing observations.
NMISS	specifies that observations are accumulated based on the number of missing observations.
NOBS	specifies that observations are accumulated based on the number of observations.
FIRST	specifies that observations are accumulated based on the first of their values.
LAST	specifies that observations are accumulated based on the last of their values.
STDDEV | STD	specifies that observations are accumulated based on the standard deviation of their values.
CSS	specifies that observations are accumulated based on the corrected sum of squares of their values.
USS	specifies that observations are accumulated based on the uncorrected sum of squares of their values.

Here are some examples:

- To use an average accumulation for all variables, specify **`ACCUMULATE=AVG`**.
- To use a total accumulation for all variables except for var1, which will use an average accumulation, specify **`ACCUMULATE= TOTAL AVG(var1)`**.
- To use a total accumulation for all variables except var1 and var2, which will use an average accumulation, and var3, which will accumulate the observations based on standard deviations, specify **`ACCUMULATE=TOTAL AVG(var1 var2) STD(var3)`**.

ADJUST= *var1*(*var2var3*) [/operation=(pre,post)]
Specifies the adjustment options for the dependent variables. If not specified, the default value for pre-adjust and post-adjust is NONE.

Here are the valid pre-adjust and post-adjust values:

- NONE
- ADD
- SUBTRACT
- MULTIPLY
- DIVIDE
- MIN
- MAX

Here are some examples:

- To adjust the sale variable with the values of the price1 variable, specify **`ADJUST=sale(price1)`**.
- To adjust the sale variable with the values of the price1 variable and to adjust the price variable with the values of the price2 variable, specify **`ADJUST=sale(price1) price(price2)`**.
- To adjust the sale variable with the values of the price1 and price2 variables, specify **`ADJUST=sale(price1 price2)`**.
- To adjust the sale variable with the values of the price1 variable before any forecasts are generated, specify **`ADJUST=sale(price1)/ operation=(ADD,NONE)`**.

AGGREGATE= TYPE [TYPE(*var1var2*) TYPE(*var3var4*) ...]
specifies the aggregation options for the dependent, input, and reporting variables. Aggregation is valid only if HIERARCHY=YES. See the ACCUMULATE option for valid values and examples.

Note: For the dependent variable, the only valid values are TOTAL | AVERAGE | AVG.

ALLOWNEGATIVE= YES | TRUE | NO | FALSE
specifies whether negative forecasts are allowed. The default value is NO.

ALPHA= *n*
specifies the confidence level for the series. The default value is 0.05, which is a 95% confidence level.

ARIMAX= YES | TRUE | NO | FALSE
specifies whether ARIMA models should be considered. The default value is YES.

BACK= *n*
specifies the out-of-sample range (the number of periods from the end). The default value is 0.

BY= *variable*
specifies the BY variables. Multiple variable names must be separated by a space.

COMBINE = YES | TRUE | NO | FALSE
specifies whether SAS Forecast Server should automatically generate a combined model for the set of time series models that is generated from the diagnosis of each time series. The default value is **`NO`**.

COMBINECRITERION = *selection-criterion*
specifies the statistic of fit to use when ranking the combined models. The default value is **`RMSE`**.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMAX option, the UCM option, or both the ARIMAX and UCM options are set to YES or TRUE.

Here are the possible values for the selection criterion:

- AADJRSQ — Amemiya's adjusted R-squared
- ADJRSQ — adjusted R-squared
- AIC — Akaike information criterion
- AICC — Akaike information criterion, finite sample size corrected
- APC — Amemiya's prediction criterion
- GMAPE — geometric mean percent error
- GMAPES — geometric mean absolute error percent of standard deviation
- GMAPPE — geometric mean predictive error
- GMASPE — geometric mean symmetric percent error
- GMRAE — geometric mean relative absolute error
- MAE — mean absolute error
- MAPE — mean absolute percent error
- MAPES — mean absolute error percent of standard deviation
- MAPPE — mean absolute predictive symmetric percent error
- MASE — mean absolute scaled error
- MAXERR — maximum error
- MAXPE — maximum percent error
- MAXPPE — maximum predictive percent error
- MAXRE — maximum relative error
- MAXSPE — maximum symmetric percent error
- MDAPE — median absolute percent error
- MDAPES — median absolute percent error percent of standard deviation
- MDAPPE — median absolute predictive percent error
- MDASPE — median absolute symmetric percent error
- MDRAE — median relative absolute errors
- ME — mean error
- MINERR — minimum error
- MINPE — minimum percent error
- MINPPE — minimum predictive percent error
- MINRE — minimum relative error

- MINSPE — minimum symmetric percent error
- MPE — mean percent error
- MPPE — mean predictive percent error
- MRAE — mean relative absolute error
- MRE — mean relative error
- MSPE — mean symmetric percent error
- MSE — mean square error
- RMSE — root mean square error
- RSQUARE — R-squared
- RWRSQ — random walk R-squared
- SBC — Schwarz Bayesian information criterion
- SMAPE — mean absolute symmetric percent error
- SSE — sum of squares error
- UMSE — unbiased mean square error
- URMSE — unbiased root mean square error

COMBINEENCOMPASS = OLS | HLN
specifies the forecast encompassing test to use to remove any models that contain redundant information. The default value is **OLS**.

Note: This option is valid only if the following conditions are met:

- the COMBINE option is set to YES or TRUE
- the ARIMA option, the UCM option, or both the ARIMA and UCM options are set to YES or TRUE

Here are the valid values

OLS uses an ordinary least squares (OLS) regression test to estimate pairwise encompassing between candidate forecasts.

HLN uses the Harvey-Leybourne-Newbold (HLN) test to estimate pairwise encompassing between candidate forecasts.

COMBINEMETHOD = AVERAGE | AICC | OLS | ERLS | LAD | NERLS | NRLS | RMSEWGT
specifies the method for determining the combination weights. These combination weights are used to calculate the weighted average of the candidate forecasts. The default method is **AVERAGE**.

Note: This option is valid only if the following conditions are met:

- the COMBINE option is set to YES or TRUE
- the ARIMA option, the UCM option, or both the ARIMA and UCM options are set to YES or TRUE

Here are the valid values:

AVERAGE computes the simple average of the forecasts that you selected for combination. This is the default.

AICC uses the Akaike's information criterion to compute the combination weights based on corrected AIC weights.

OLS — uses ordinary least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast residuals.

ERLS — uses equally restricted least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast subject to the constraint that the weights sum to 1.

LAD — uses a measure of fit based on the least absolute deviation to compute the combination weights.

NERLS — uses nonnegative, equality restricted least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast residuals subject to the constraints that the weights sum to 1 and be nonnegative.

NRLS — uses nonnegative restricted least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast residuals subject to the constraints that the weights be nonnegative.

RANKWGT — assigns weights using the rank of the candidate forecasts at the time the combination is performed. The weights must sum to 1. If not, the last value in the list is adjusted, and a warning is issued. The weights are assigned by ranking the candidate forecasts from best to worst. The best uses the first weight, and so on. The set of weights used is normalized to account for candidates that fail to forecast or for candidates that are omitted from the final combination because of any exclusion tests.

RMSEWGT — uses the weights of the root mean square error to compute the combination weights. The weights are normalized to sum to 1.

COMBINEMISSMODE = MISSING | RESCALE

specifies the method for treating missing values in the forecast combination. In a given time series across the combination ensemble, one or more combination contributors can have a missing value. This setting determines the treatment of those in the final combination for such time indices. The default value is **`MISSING`**.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMAX option, the UCM option, or both the ARIMAX and UCM options are set to YES or TRUE.

MISSING — generates a missing combined forecast at each time index with one or more missing contributors. This method is the default for AICC weights, OLS weights, restricted least squares weights, and LAD weights. You cannot rescale the combination weights if you selected the OLS-based regression test to estimate pairwise encompassing. In this test, the estimated weights are not constrained to sum to one.

RESCALE — rescales the combination weights for the nonmissing contributors at each time index to sum to 1. This method is the default for simple average, user-specified weights, ranked user weights, ranked weights, and root mean square error (RMSE) weights.

COMBINEMISSPCT = *integer*
specifies a threshold for the percentage of missing values in the combination estimation region. This threshold is used to exclude a candidate forecast from consideration in the final combination. By default, no missing percentage test is performed on candidate forecasts. Valid values are 1 to 100.

The COMBINEMISSPCT option and the HORMISSPERCENT option are independent of each other. You can specify one or both of these options.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMAX option, the UCM option, or both the ARIMAX and UCM options are set to YES or TRUE.

COMBINESTDERR = DIAG | ESTCORR
specifies the method for computing the prediction error variance series. This series is used to compute the prediction standard error, which in turn is used to compute confidence bands on the combined forecast. The default value is **`DIAG`**.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMA option, the UCM option, or both the ARIMA and UCM options are set to YES or TRUE.

CRITERION= *options*
specifies the model selection criterion (statistic of fit) to be used to select from several candidate models. The default value is MAPE. For more values, see the COMBINECRITERION option.

DESCRIPTION= *text*
specifies the project description.

DETECTOUTLIERS= YES | TRUE | NO | FALSE
specifies whether outliers in the data should be detected when fitting an ARIMA model. The default value is **`NO`**.

DISAGGREGATION= PROPORTIONS | EQUALSPLIT
specifies the disaggregation method for reconciliation. The default value is **`PROPORTIONS`**.

ENCOMPASSALPHA=*value*
specifies the significance level (or alpha value) for the forecast encompassing test. Valid values are 0 to 1 (inclusive). The default value is **`0.05`**. See also the COMBINEENCOMPASS option.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMA option, the UCM option, or both the ARIMA and UCM options are set to YES or TRUE.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

ESM= YES | TRUE | NO | FALSE
specifies whether exponential smoothing models (ESM) should be considered. The default value is **`YES`**.

ESMONLYATLEVEL= *BY- variable*
for hierarchical projects, specifies the highest level to fit the exponential smoothing models (ESM). Use this option to improve performance for projects with large hierarchies. The argument specified must be the name of a BY variable. By default, no level is specified for fitting the ESM models.

HIERARCHY= YES | TRUE | NO | FALSE
specifies whether the BY variables should be considered hierarchical. The default value is **NO**.

HOLDOUT= *n* | NO | FALSE
specifies the number of periods to be used as the holdout sample for model selection. If you do not want to specify a holdout sample, set this option to NO or FALSE.

HORMISSPCT=*integer*
specifies a threshold for the percentage of missing forecast values in the combination horizon. This threshold is used to exclude a candidate forecast from consideration in the final combination. Valid values range from 1 to 100 (inclusive).

This option is different from the COMBINEMISSPCT option, which is for the estimation region. The COMBINEMISSPCT option and the HORMISSPERCENT option are independent of each other. You can specify one or both of these options.

Note: This option is valid only if the following conditions are met:

- The COMBINE option is set to YES or TRUE.
- The ARIMA option, the UCM option, or both the ARIMA and UCM options are set to YES or TRUE.

HOLDOUTPCT= *n* | NO | FALSE
specifies the number of periods as a percentage of the series length to be used as the holdout sample for model selection. If you do not want to specify a holdout sample, set this option to NO or FALSE.

IDFORMAT= *date*
specifies the date-and-time format of the values in the time ID variable.

INPUT= *variable*
specifies the input (or independent) variables. Separate multiple variable names by a space.

INTERMITTENT= *n* | NO | FALSE
specifies a number greater than 1 that is used to determine whether a time series is intermittent. The default value is **2**. To disable this test, set this option to NO or FALSE.

INTERVAL= *interval-measure*
specifies the time interval of the time ID variable. If a time interval is not specified, SAS Forecast Server detects the time interval from the data.

Here is the list of valid values:

- for SAS date values: YEAR, SEMIYEAR, QTR, MONTH, SEMIMONTH, TENDAY, WEEK, WEEKDAY, DAY, YEARV, R445YR, R454YR, R544YR, R445QTR, R454QTR, R544QTR, R445MON, R454MON, R544MON, and WEEKV
- for SAS datetime values: HOUR, MINUTE, and SECOND

Note: Add a DT prefix to any of the date intervals to create the datetime interval.

Abbreviations or aliases of these intervals are not valid values. You can add multiplier and shift parameters to any of these intervals. For the WEEKDAY

interval, you can add values for the weekend day. Custom interval names are also allowed. For more information, see "Time Intervals" in the *SAS/ETS User's Guide*.

LEAD= *n*
specifies the number of periods into the future in which multiple step forecasts are made. The larger the horizon value, the larger the prediction error variance at the end of the horizon. The default value is **12**.

MAXNUMOUTLIERS= *n*
specifies the maximum number of outliers to include in ARIMAX models. The default value is **2**.

MAXPCTOUTLIERS= *n*
specifies the maximum number of outliers to include in ARIMAX models as a percentage of the series length, not including beginning and ending missing values. The default value is **2**.

If you specify both the MAXNUMOUTLIERS and MAXPCTOUTLIERS options, then SAS Forecast Server uses the smaller value. For example, you set MAXNUMOUTLIERS=5 and MAXPCTOUTLIERS=10. The maximum number of the outliers is 5 or 10% of the series length, whichever is smaller.

MINOBSTREND= *n*
specifies the minimum number of observations needed for a trend model. The default value is **2**.

MINOBSNONMEAN= *n*
specifies the minimum number of observations needed for a non-mean model. The default value is 1.

MINOBSSEASON= *n*
is used to determine the number of observations required for a seasonal model. The value of the MINOBSSEASON option is multiplied by the seasonal cycle length. In order for a seasonal model to be fit to a series, the model must have more observations than the calculated value. Models with fewer observations will not be fit.

The value of MINOBSSEASON must be greater than or equal to 1. The default value is **2**.

MODELSELECTIONLIST= *SAS-data-set*
specifies a model selection list that contains models that you can use. The default value is **SASHELP.HPFDFLT.TSFSSELECT**.

PUBLICACCESS= YES | TRUE | NO | FALSE
specifies whether the project can be opened by other users. The default value is **NO**.

RECONCILIATION= TOPDOWN | TD | BOTTOMUP | BU | MIDDLEOUT(*BY-variable*) | MO(*BY-variable*) | NO | FALSE
specifies whether SAS Forecast Server reconciles the hierarchy. By default, SAS Forecast Server reconciles the hierarchy using the top-down reconciliation method (TOPDOWN or TD). To reconcile the hierarchy by a middle level, you must specify the BY variable for the level where the reconciliation starts.

Here are some examples of how to specify this option:

- To specify a top-down reconciliation, use RECONCILIATION=TOPDOWN.
- To specify a middle-out reconciliation that starts at the regionName variable, use RECONCILIATION=MIDDLEOUT(regionName).
- If you do not want to reconcile the hierarchy, specify RECONCILATION=NO (or FALSE).

REPORTING= *variable*
specifies the reporting (or auxiliary) variables for the project. Separate multiple variable names by a space.

RUN= YES | TRUE | NO | FALSE
specifies whether to produce forecasts when the project is created. The default value is **`YES`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

SEASONALITY= *n*
specifies the length of a season. When possible, SAS Forecast Server determines the seasonal cycle length from the time ID variable. You can specify a seasonal cycle length other than the default if you want to model a cycle in the data. For example, if your data contains a 13-week cycle, then use SEASONALITY=13.

SEASONTEST= *n* | NO | FALSE
specifies the sensitivity of the seasonality test. You can specify no seasonality test or specify a significance probability value for the test. Series with strong seasonality have small test probabilities. A significance probability value of 0 always implies seasonality. A significance probability value of 1 always implies no seasonality.

The default value is **`0.01`**. To disable the test, set this option to NO or FALSE.

SETMISSING= 0 | MISSING | AVG | MIN | MED | MAX | FIRST | LAST | PREV | NEXT
specifies how to replace the missing values in the data. The default value is **`MISSING`**.

- 0 - The missing values are set to 0.
- MISSING - The missing values are set to missing. This is the default value.
- AVG - The missing values are set to the accumulated average value.
- MIN - The missing values are set to the accumulated minimum value.
- MED - The missing values are set to the accumulated median value.
- MAX - The missing values are set to the accumulated maximum value.
- FIRST - The missing values are set to the accumulated first nonmissing value.
- LAST - The missing values are set to the accumulated last nonmissing value.
- PREV - The missing values are set to the previous accumulated nonmissing value. Missing values at the beginning of the accumulated series remain missing.
- NEXT - The missing values are set to the next accumulated nonmissing values. Missing values at the end of the accumulated series remain missing.

START= *date*
specifies the first date to use in data preparation. The date format is ddmmmyyyy and the date-and-time format is ddmmmyyyy:hh:mm:ss. The default value is the earliest date in the input data set.

TIMEMULTIPLIER= *n*
specifies the multiplier for the time interval that you specified by using the INTERVAL= option. You could have specified the multiplier in the INTERVAL option. For example, if you specified INTERVAL=WEEK2, then the time interval is WEEK and the multiplier is 2.

TIMESHIFT= *n*
specifies the offset for the time interval that you specified in the INTERVAL= option. You could have specified this offset in the INTERVAL option. For example, if you specified INTERVAL=WEEK.2, then the time interval is WEEK, and the time shift is 2 periods.

TRANSBOXCOX= *n*
specifies the Box-Cox value if the transformation type (specified by the TRANSTYPE option) is BOXCOX. Valid values range from -5 to 5 (inclusive). The default is **1**.

TRANSOPT= MEAN | MEDIAN
specifies how to calculate the forecasts for the system-generated models. The default is MEAN.

TRANSTYPE= AUTO | LOG | NONE | SQRT | LOGISTIC | BOXCOX
specifies the transformation to apply to the dependent variable or independent variable. This transformation applies to the system-generated models.

Here is a description for each valid value:

AUTO	automatic transformation. SAS Forecast Studio uses the model selection criteria to determine whether to apply a logarithmic transformation or no transformation.
LOG	logarithmic transformation. This option is not available for negative series.
NONE	no transformation. This is the default.
SQRT	square-root transformation.
LOGISTIC	logistic transformation.
BOXCOX	Box-Cox transformation. If you select this type of transformation, then you can specify a value for this transformation using the TRANSBOXCOX option.

TRIMMISS= NONE | LEFT | RIGHT | BOTH
specifies how missing values are removed from the accumulated time series. The default value is **NONE**.

Here is a description of each valid value:

NONE	The missing values are kept.
LEFT	The beginning missing values are removed.
RIGHT	The ending missing values are removed.
BOTH	Both the beginning and ending missing values are removed.

UCM= YES | TRUE | NO | FALSE
specifies whether SAS Forecast Server should automatically generate an unobserved component model (UCM). The default value is **NO**.

WEEKENDDAYS= 1 | 2 | 3 | 4 | 5 | 6 | 7
specifies which days are the weekend (or inactive) days in the week. Only valid when INTERVAL=WEEKDAY. The days are specified as numbers (1 through 7) representing the days of the week (Sunday through Saturday).

Here are some examples:

- To specify Saturday and Sunday as the weekend (or inactive) days in the week, use WEEKENDDAYS=17

- To specify Friday and Saturday as the weekend days, use WEEKENDDAYS=67.

The weekend (or inactive) days can also be specified using the INTERVAL option. For example, INTERVAL=WEEKDAY67W is the same as setting INTERVAL=WEEKDAY and WEEKENDDAYS=67.

ZEROMISS= NONE | LEFT | RIGHT | BOTH
specifies how beginning or ending zero values (or both) are interpreted in the accumulated time series. The default value is **NONE**.

Here is a description of each valid value:

NONE	The beginning and ending zeros are unchanged.
LEFT	The beginning zeros are set to missing.
RIGHT	The ending zeros are set to missing.
BOTH	Both the beginning and ending zeros are set to missing.

Results

The FSCREATE global macro variable indicates whether the FSCREATE macro finishes successfully or encounters errors: &FSCREATE = SUCCESS | ERROR.

A series of global macro variables are populated after a SAS Forecast Server project is loaded. These macro variables describe the project. Using these macro variables, you can write SAS code that is generic enough to use across multiple projects. For example, you can use this code to explore and use data in your SAS Forecast Server projects. You can also use these macro variables in SAS Stored Processes.

Example

```
%fscreate(projectname=pd1,
user=sasdemo,
password=Password1,
environment=Default,
data=sashelp.pricedata,
id=date,
by=regionName productLine productName,
hierarchy=YES,
var=sale,
input=price discount,
reporting=price1 price2 price3,
accumulate=TOTAL AVG(price) NONE(price1 price3),
aggregate=NONE total(sale) MIN(price1 price3),
disaggregation=EQUALSPLIT,
reconciliation=NO,
publicaccess=YES,
setmissing=last,
zeromiss=left,
trimmiss=right,
back=1,
start=01Feb1998,
interval=MONTH,
seasonality=12,
idformat=MMYY.,
timemultiplier=1,
timeshift=1,
```

```
weekenddays=45,
arimax=YES,
esm=YES,
modelselectionlist=sashelp.hpfdflt.tsfsselect,
detectoutliers=YES,
intermittent=NO,
seasontest=0.3,
holdout=NO,
holdoutpct=2,
minobstrend=2,
minobsnonmean=2,
criterion=MAPE,
lead=24,
alpha=0.10,
allownegative=NO,
description=foo bar stuff,
transopt=MEDIAN,
transtype=boxcox,
transboxcox=2
);
```

FSDELARC Macro

The FSDELARC macro deletes an archived SAS Forecast Server project.

Syntax

%FSDELARC (PROJECTNAME= , ARCHIVENAME= , USER= , PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSDELARC macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

ARCHIVENAME= *archived-project-name*
specifies the filename of the archive that you want to delete the project from (for example, myarchive.far). If the archive name does not end with .far, then the extension is automatically assumed.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

The following options can be used with the FSDELARC macro. Options must follow the required arguments and are separated by commas.

ARCHIVEFOLDER= *archive-folder-name*
specifies the directory containing the archive to delete. If not specified, the default location is assumed.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **`NO`**.

Results

The FSDELARC global macro variable indicates whether the FSDELARC macro finishes successfully or encounters errors: &FSDELARC = SUCCESS | ERROR.

Example

```
%fsdelarc(projectname=pd1,
archivename=ArchPD1.far,
user=sasuser,
password=saspass,
sasEnvironment=default,
nowarn=NO
);
```

FSDELENV Macro

The FSDELENV macro deletes an environment. When you delete an environment, the metadata, and file system content are deleted. You must be an administrator to delete an environment.

Syntax

%FSDELENV (ENVIRONMENT= , USER= , PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSDELENV macro. The required arguments are separated by commas.

ENVIRONMENT= *project-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSDELENV global macro variable indicates whether the FSDELENV macro finishes successfully or encounters errors: &FSDELENV = SUCCESS | ERROR.

Example

```
%fsdelenv(enviroment=Default,
user=sasuser,
password=saspass,
sasenvironment=default
);
```

FSDELEVT Macro

The FSDELEVT macro deletes selected events from the event repository.

Syntax

%FSDELEVT(PROJECTNAME=,EVENTNAMES=,USER=,PASSWORD, [, options]);

Details

Required Arguments

The following arguments must be used with the FSDELEVT macro. The required arguments are separated by commas.

PROJECTNAME= *text*
specifies the name of the product.

EVENTNAMES= *name1 name2*
specifies the events that you want to delete. Only one event name is required. Separate multiple names by a space. Event names are case sensitive.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

The following options can be used with the FSDELEVT macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *project-name*
specifies the name of the SAS Forecast Server environment. The default environment is **Default**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

EVENTINTEGRITYRULE= NONE | CHECK | FORCE
determines how events are deleted when those events are in a combination event or model selection lists. The default value is **NONE**.

NONE	No integrity checks are performed. The events are removed from the event repository unconditionally.
CHECK	SAS Forecast Server checks combination events and model selection lists for these events. An error appears if any of the events are found.
FORCE	The events are removed from combination events, model selection lists, and the event repository.

Results

The FSDELEVT global macro variable indicates whether the FSDELEVT macro terminates successfully or encounters errors: &FSDELEVT= SUCCESS | ERROR.

Example

```
%fsdelevt(projectname=Project1,
         eventnames=evt2 evt4,
         user=sasuser,
         password=saspass,
         environment=Default
         );
```

FSDELPRJ Macro

The FSDELPRJ macro deletes one project, and if you choose, any related archives in the default archiving folder.

Syntax

%FSDELPRJ (PROJECTNAME= , USER= , PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSDELPRJ macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

The following options can be used with the FSDELPRJ macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

DELETEARCHIVES= YES | TRUE | NO | FALSE
specifies whether the project archives in the default archive location are to be deleted. By default, all archives are deleted with the project.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **`NO`**.

Results

The FSDELPRJ global macro variable indicates whether the FSDELPRJ macro finishes successfully or encounters errors: &FSDELPRJ = SUCCESS | ERROR.

Example

```
%fsdelprj(projectname=prdat4,
user=sasuser,
password=saspass
);
```

FSEVTREQ Macro

The FSEVTREQ macro sets the required attributes on events in the project's event repository.

Syntax

%FSEVTREQ (PROJECTNAME=, EVENTNAME=, EVENTREQUIRED=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSEVTREQ macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

EVENTNAME=*event-name*
specifies the name of the event. To set attributes for all events in the project's event repository, use the _ALL_ keyword in place of *event-name*.

EVENTREQUIRED= YES | MAYBE | NO | UNDEF
specifies whether the event must be included in the model.

Here are the descriptions of the valid values:

YES	specifies to include the event in the model as long as the model can be diagnosed.
MAYBE	specifies to include the event in the model if the parameters of the event are significant.
NO	specifies to include the event in the model if the parameters of the event are significant and the value of the criterion exceeds a specified threshold.
UNDEF	specifies not to include the event in the model.

USER= *user-name*
specifies the administrative user name that you use to log on to SAS Forecast Studio. The user must be a SAS administrator.

PASSWORD= *password*
specifies the administrative user password that you use to log on to SAS Forecast Studio.

Options

The following options can be used with the FSEVTREG macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSEVTREG global macro variable indicates whether the FSEVTREG macro finishes successfully or encounters errors: &FSEVTREG = SUCCESS | ERROR.

Example

```
%fsevtreg(projectname=prdata1,
eventname=_ALL_,
eventrequired=undef,
user=sasuser,
password=saspass,
sasenvironment=default
);
```

FSEXPALL Macro

The FSEXPALL macro exports all SAS Forecast Server projects to archived files. This macro should be used only by users who are assigned the Administer Product capability.

Syntax

%FSEXPALL (USER=, PASSWORD=, OUT= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSEXPALL macro. The required arguments are separated by commas.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio. The user must be a SAS administrator.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

OUT= *SAS-data-set*
specifies the fully qualified name of the data set that contains information about the archived projects. The format is *library.dataset.*

Options

The following options can be used with the FSEXPALL macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The value for this argument is case sensitive. The default environment is default.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

NAMESUFFIX= *string*
specifies a suffix to the name of the archive. The suffix must make the archive name unique. If an archive with the same name exists, then archiving fails for that project. The default suffix is **`_fs_export`**.

CPORT= YES | TRUE | NO | FALSE
exports data sets and catalogs using CPORT. Use this argument if you unarchive the project on a different platform. The default is **NO**.

ARCHIVEFOLDER= *directory-name*
specifies the directory where the archive is to be saved. If not specified, the archives are saved to project subfolders in the default location for the environment.

SUBFOLDERS= YES | TRUE | NO | FALSE
specifies whether archives are saved in subfolders corresponding to project names. If NO, all archives are saved in the same location. To use this option, you must specify the ARCHIVEFOLDER option. The default is **NO**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **NO**.

Results

The FSEXPALL global macro variable indicates whether the FSEXPALL macro finishes successfully or encounters errors: &FSEXPALL = SUCCESS | ERROR.

The OUT= option produces a SAS data set that contains the following variables:

NAME	specifies the project name.
CREATED	specifies the date-and-time when the project is created.
MODIFIED	specifies the date-and-time when the project is last modified.
CREATEDBY	specifies the user ID that created the project initially.
OWNEDBY	specifies the user ID of the project's owner.
LOCKED	equals 1 if the project is locked (currently opened by another user).
CANOPEN	equals 1 if the user can open the project.
CANDELETE	equals 1 if the user can delete the project.
ISPUBLIC	equals 1 if the project has public access.
ARCHIVENAME	specifies the name of the exported project’s archive file.
ARCHIVEFAIL	equals 1 if an error was detected during archiving the project.

In addition, archives for all projects are created in the default archive folder. The name of the archives are the name of the project followed by the NAMESUFFIX= parameter.

Example

```
%fsexpall(out=work.projects,
user=sasadm,
password=sasadmpass,
namesuffix=_fs_export31,
);
```

FSEXPORT Macro

The FSEXPORT macro exports a SAS Forecast Server project to an archive file.

Syntax

%FSEXPORT (PROJECTNAME=, ARCHIVENAME=, DESCRIPTION=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSEXPORT macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

ARCHIVENAME= *archived-project-name*
specifies the filename of the archive that you want to export the project into (for example, myarchive.far). If the archive name does not end with .far, then the extension is automatically assumed. Use the ARCHIVEFOLDER option to specify a file path to the archive file.

DESCRIPTION= *text*
specifies a description to assign to the archive. The description is recorded in the Manifest.fs file within the archive. If a comma is present in the description string, the description must be enclosed in quotation marks.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

The following options can be used with the FSEXPORT macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

ARCHIVEFOLDER= *directory-name*
specifies the directory where the archive is to be saved. If not specified, the default location is used.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **`NO`**.

CPORT= YES | TRUE | NO | FALSE
exports data sets and catalogs using CPORT. Use this argument if you unarchive the project on a different platform. The default is **`NO`**.

Results

The global macro variable FSEXPORT indicates whether the FSEXPORT macro finishes successfully or encounters errors: &FSEXPORT = SUCCESS | ERROR.

Also, an archived project with the ARCHIVENAME= filename is created in the default archive folder.

Example

```
%fsexport( projectname=pd1,
archivename=ArchPD1.far,
description=Project pd1,
user=sasuser,
password=saspass
);
```

FSGETENV Macro

The FSGETENV macro retrieves information about the SAS Forecast Server environments. Only an administrative user can run this macro.

Syntax

%FSGETENV (OUT=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSGETENV macro. The required arguments are separated by commas.

OUT= *SAS-data-set*
specifies the name of the data set that contains the program results.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSGETENV macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSGETENV global macro indicates whether the FSGETENV macro finishes successfully or encounters errors: &FSGETENV = SUCCESS | ERROR.

The OUT= option produces a SAS data set that contains the following variables:

NAME	specifies the environment name.
DESCRIPTION	specifies the description of the environment.
CREATED	specifies the date-and-time when the environment was created.
MODIFIED	specifies the date-and-time when the environment was last modified.
HOSTNAME	specifies the name of the workspace server used by the environment.
REPORTSROOT	specifies the SAS metadata folder where the stored process metadata objects are stored.
CONTENTPATH	specifies the file system path associated with the environment.
VERSION	specifies the version of SAS Forecast Server that the environment uses.

Example

```
%fsgetenv(out=work.getprojects,
user=sasuser,
password=saspass,
sasenvironment=default
);
```

FSGETPRJ Macro

The FSGETPRJ macro retrieves information about the SAS Forecast Server projects.

Syntax

%FSGETPRJ (OUT=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSGETPRJ macro. The required arguments are separated by commas.

OUT= *SAS-data-set*
specifies the name of the data set that contains the program results.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSGETPRJ macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

PRINTVERSIONONLY= 0 | 1
specifies whether to print the version number of the project to the log. No project information is generated.

Results

The FSGETPRJ global macro indicates whether the FSGETPRJ macro finishes successfully or encounters errors: &FSGETPRJ=SUCCESS | ERROR.

The OUT= option produces a SAS data set that contains the following variables:

NAME	specifies the project name.
CREATED	specifies the date-and-time when the project is created.
MODIFIED	specifies the date-and-time when the project is last modified.
CREATEDBY	specifies the user ID that created the project initially.
OWNEDBY	specifies the user ID of the project's owner.
LOCKED	equals 1 if the project is locked (currently opened by another user).
CANOPEN	equals 1 if the user can open the project.
CANDELETE	equals 1 if the user can delete the project.
ISPUBLIC	equals 1 if the project is available to all users.

Example

```
%fsgetprj(out=work.getprojects,
user=sasuser,
password=saspass
);
```

FSGETURP Macro

The FSGETURP macro creates a list of unregistered project names in a specified environment. To register the projects, you can use the project names generated by the FSGETURP macro as input to the FSREGPRJ macro.

Syntax

%FSGETURP (USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSGETURP macro. The required arguments are separated by commas.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSGETURP macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The value for this argument is case sensitive. The default environment is default.

Results

The FSGETURP global macro variable indicates whether the FSGETURP macro finishes successfully or encounters errors: &FSGETURP = SUCCESS | ERROR.

Example

```
%fsgeturp(user=sasadm,
password=Password1,
environment=Default
);
```

FSIMPALL Macro

The FSIMPALL macro imports all of the SAS Forecast Server archived files listed in the PROJECTDS= data set. This macro should be used only by users who have the Administer Product capability.

Syntax

%FSIMPALL (PROJECTDS=, ARCHIVEFOLDER=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSIMPALL macro. The required arguments are separated by commas.

PROJECTDS= *SAS-data-set*
specifies the SAS data set that contains the list of the SAS Forecast Server projects to import. It can be a fully qualified data set name in the form library.dataset. The PROJECTDS= data set is created when you specify the OUT= option of the FSEXPALL macro.

The PROJECTDS= data set must be a SAS data set that contains the following variables:

Name
specifies the project name.

Archive Name
specifies the name of the archive.

Archive Fail
(optional) equals 1 if an error was detected during archiving. If the Archive Fail variable is included, only the archives where the value of the Archive Fail variable is 0 are imported.

ARCHIVEFOLDER= *directory-name*
specifies the folder that contains the archived projects (for example, **`C:\SAS\ForecastStudio\Archives)`**.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSIMPALL macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The value for this argument is case sensitive. The default environment is default.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **`NO`**.

SUBFOLDERS= YES | TRUE | NO | FALSE
specifies whether archives are stored in subfolders corresponding to project names within the specified archive folder location. If NO, all archives are assumed to be located in the archive folder location. The default value is **`NO`**.

Results

The FSIMPALL global macro variable indicates whether the FSIMPALL macro finishes successfully or encounters errors: &FSIIMPALL = SUCCESS | ERROR.

Note: FSIMPALL does not migrate 2.1 projects to the current version. However, you can use the FSMIGALL macro to accomplish that. For an example that uses FSIMPALL, see "Example 1: Migrating SAS Forecast Server 2.1 Projects to SAS Forecast Server 4.1" on page 158.

Example

```
%fsimpall(projectds=work.projects,
archivefolder=\\sourceserver\SAS\ForecastStudio\Archives,
user=sasdemo,
password=Password1
);
```

FSIMPORT Macro

The FSIMPORT macro imports a SAS Forecast Server archived file.

Syntax

%FSIMPORT (PROJECTNAME=, ARCHIVEPATH=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSIMPORT macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

ARCHIVEPATH= *directory-name*
specifies the location of the archived project.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSIMPALL macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **Default**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default is **NO**.

Results

The FSIMPORT global macro variable indicates whether the FSIMPORT macro finishes successfully or encounters errors: &FSIMPORT = SUCCESS | ERROR.

Example

```
%fsimport(projectname=prdat2,
archivepath=C:\SAS\ForecastStudio\Archives\pd1\ArchPD1.far,
user=sasuser,
password=saspass
);
```

FSLOAD Macro

The FSLOAD macro opens an existing SAS Forecast Server project, and loads global macro variables that describe the project.

Syntax

%FSLOAD (PROJECTNAME=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSLOAD macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSLOAD macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **Default**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSLOAD global macro variable indicates whether the FSLOAD macro finishes successfully or encounters errors: &FSLOAD = SUCCESS | ERROR.

A series of global macro variables are populated after a SAS Forecast Server project is loaded. These macro variables describe the project. Using these macro variables, you can write SAS code that is generic enough to use across multiple projects. For example, you can use this code to explore and use data in your SAS Forecast Server projects. You can also use these macro variables in SAS Stored Processes. For more information, see "What Is an Internal Report Parameter?" on page 83.

Example

```
%fsload(projectname=pd1,
user=sasuser,
password=saspass
);
```

FSMIGALL Macro

The FSMIGALL macro updates all registered projects to the current version of SAS Forecast Server.

Note: This macro updates only projects created with SAS Forecast Server 2.1. You cannot use this macro to update projects created with SAS Forecast Server 1.4 directly to SAS Forecast Server 3.1. You must first upgrade these projects to use SAS Forecast Server 2.1.

Syntax

%FSMIGALL (USER=, PASSWORD=, OUT= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSMIGALL macro. The required arguments are separated by commas.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

OUT= *SAS-data-set*
specifies the fully qualified name of the data set that will contain information about the archived projects. The format is library.dataset.

Options

You can use the following options with the FSMIGALL macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

Results

The FSMIGALL global macro indicates whether the FSMIGALL macro finishes successfully or encounters errors: &FSMIGALL=SUCCESS | ERROR.

The OUT= option produces a SAS data set that contains the following variables:

NAME	specifies the project name.
CREATED	specifies the date-and-time when the project is created.
MODIFIED	specifies the date-and-time when the project is last modified.
CREATEDBY	specifies the user ID that created the project initially.
OWNEDBY	specifies the user ID of the project's owner.
LOCKED	equals 1 if the project is locked (currently opened by another user).
CANOPEN	equals 1 if the user can open the project.
CANDELETE	equals 1 if the user can delete the project.
ISPUBLIC	equals 1 if the project is available to all users.
ISMIGRATED	equals 1 if the project was successfully migrated.

Example

```
%fsmigall(user=sasdemo,
password=Password1,
out=work.migrated,
);
```

FSMIGPRJ Macro

The FSMIGPRJ macro updates an existing SAS Forecast Server project to the current version of SAS Forecast Server.

Note: This macro updates only projects created with SAS Forecast Server 2.1. You cannot use this macro to update projects created with SAS Forecast Server 1.4 directly to SAS Forecast Server 4.1. You must first upgrade these projects to use SAS Forecast Server 2.1.

Syntax

%FSMIGPRJ (PROJECTNAME=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSMIGPRJ macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSMIGPRJ macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The value for this argument is case sensitive. The default environment is default.

Results

The FSMIGPRJ global macro indicates whether the FSMIGPRJ macro finishes successfully or encounters errors: &FSMIGPRJ=SUCCESS | ERROR.

Example

```
%fsmigprj(projectname=pd1,
user=sasuser,
password=saspass
);
```

FSMOVE Macro

The FSMOVE macro moves a SAS Forecast Server project to a new destination.

Syntax

%FSMOVE (SOURCEPROJECTNAME=, DESTINATIONPROJECTNAME=, USER=, PASSWORD=, ARCHIVEPATH=, REMOTEARCHIVEFOLDER= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSMOVE macro. The required arguments are separated by commas.

SOURCEPROJECTNAME= *source-project-name*
specifies the name of the SAS Forecast Server project in the source environment. The name must be a valid SAS name.

DESTINATIONPROJECTNAME= *destination-project-name*
specifies the name of the SAS Forecast Server project in the destination environment. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

REMOTEARCHIVEFOLDER= *project-name*
specifies the folder location of the archived project. This location must be accessible by the destination environment — for example, **`\\remotehost\SAS \ForecastStudio\Archives`**.

SOURCEENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment to be used as the source environment. This option is required only if there are multiple environments available to the user.

DESTINATIONENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment to be used as the destination environment. This option is required only if there are multiple environments available to the user.

Options

You can use the following options with the FSMOVE macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

NOWARN= YES | NO
suppresses errors if the SAS Forecast Server project does not exist. The default is **`NO`**.

ARCHIVEFOLDER= *directory-name*
specifies the directory to save the archive in. If not specified, the default location is used.

TMPARCHIVENAME= *value*
specifies the name of the temporary archived project. The default is **`_fs_tmp_archive`**. The name must not match an existing archive.

CPORT= YES | TRUE | NO | FALSE
exports data sets and catalogs using CPORT. Needed if the source and destination are running on different operating systems. The default is **`NO`**.

Results

The FSMOVE global macro indicates whether the FSMOVE macro finishes successfully or encounters errors: &FSMOVE=SUCCESS | ERROR.

Example

```
%fsmove(sourceprojectname=prdat1,
destinationprojectname=prdat2,
user=sasadm,
password=saspass,
remoteArchiveFolder=C:\SAS\ForecastStudio\Archives,
sourceEnvironment=Default,
destinationEnvironment=Default,
);
```

FSNEWENV Macro

The FSNEWENV macro creates a new SAS Forecast Server environment.

Note: Only an administrative user can run this macro.

Syntax

%FSNEWENV (ENVIRONMENT=, USER=, PASSWORD=, PATH=, REPORTSPATH= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSNEWENVJ macro. The required arguments are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the new SAS Forecast Server environment.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

PATH= *file path*
specifies the absolute file path to the base content directory.

REPORTSPATH= *UNIX-style path*
specifies a UNIX-style absolute path to the Base SAS Folder in the metadata (for example, **`/MyContent/Reports`**). Searches for reports start in this directory.

Options

You can use the following options with the FSNEWENV macro. Options must follow the required arguments and are separated by commas.

DESC= *text*
specifies the environment description.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

WSSERVER= *workspace server-name*
specifies the name of the logical workspace server. The default value is **SASAPP - Logical Workspace Server**.

Results

The FSNEWENV global macro indicates whether the FSNEWENV macro finishes successfully or encounters errors: &FSNEWENV=SUCCESS | ERROR.

Example

```
%fsnewenv(environment=Default,
user=sasuser,
password=saspass,
desc=A new environment,
path=C:\FSEnvironment,
reportspath=/MyContent/Reports,
sasenvironment=default,
wsserver=SASApp - Logical Workspace Server
);
```

FSREGENV Macro

The FSREGENV macro registers an existing directory structure as an environment and registers all projects in that directory.

Note: Only an administrative user can run this macro.

Syntax

%FSREGENV (ENVIRONMENT=, USER=, PASSWORD=, PATH=, REPORTSPATH= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSREGENV macro. The required arguments are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the new SAS Forecast Server environment.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

PATH= *file path*
specifies the absolute file path to the base content directory on the specified workspace server.

REPORTSPATH= *UNIX-style path*
specifies a UNIX-style absolute path to the Base SAS Folder in metadata (for example, **/MyContent/Reports**). Searches for reports start in this directory.

Options

You can use the following options with the FSREGENV macro. Options must follow the required arguments and are separated by commas.

DESC= *text*
specifies the environment description.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

WSSERVER= *workspace server-name*
specifies the name of the logical workspace server. The default value is SASAPP - Logical Workspace Server.

REGISTERPROJECTS= YES | TRUE | NO | FALSE
specifies whether to register in metadata any existing projects in the environment. The default value is **YES**.

Results

The FSREGENV global macro indicates whether the FSREGENV macro finishes successfully or encounters errors: &FSREGENV=SUCCESS | ERROR.

Example

```
%fsregenv(environment=Default,
user=sasuser,
password=saspass,
desc=A new environment,
path=C:\FSEnvironment,
reportspath=/MyContent/Reports,
sasenvironment=default,
wsserver=SASApp - Logical Workspace Server
registerprojects=no
);
```

FSREGPRJ Macro

The FSREGPRJ macro registers project information in metadata.

Syntax

%FSREGPRJ (USER=,PROJECTNAME=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSREGPRJ macro. The required arguments are separated by commas.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSREGPRJ macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

Results

The FSREGPRJ global macro indicates whether the FSREGPRJ macro finishes successfully or encounters errors: &FSREGPRJ=SUCCESS | ERROR.

Example

```
%fsregprj(projectname=ProjectA,
user=sasadm,
password=Password1,
environment=Default
);
```

FSREN Macro

The FSREN macro renames a SAS Forecast Server project.

Syntax

%FSREN (PROJECTNAME=, NEWPROJECTNAME=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSREN macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

NEWPROJECTNAME= *project-name*
specifies the new name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSREN macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

NOWARN= YES | NO
specifies whether to suppress errors if the SAS Forecast Server project does not exist. The default value is **`NO`**.

Results

The FSREN global macro indicates whether the FSREN macro finishes successfully or encounters errors: &FSREN=SUCCESS | ERROR.

Example

```
%fsren(projectname=prdat3,
newprojectname=prdat4,
user=sasuser,
password=saspass
);
```

FSRUNPRJ Macro

The FSRUNPRJ macro opens an existing SAS Forecast Server project and runs the project at a given stage.

Syntax

%FSRUNPRJ (PROJECTNAME=, USER=, PASSWORD=, METHOD=, IMPORTDATA= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSRUNPRJ macro. The required arguments are separated by commas.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

METHOD= CREATE | DESTRUCTIVE-DIAGNOSE | DIAGNOSE | SELECT | FIT | FORECAST | RECONCILE | OVERRIDES | CURRENT
specifies the stage when the project opens in SAS Forecast Studio.

IMPORTDATA= YES | TRUE | NO | FALSE
specifies whether to update the data in the project with any new data in the input data set. The default value is **`NO`**, and any changes to the input data source are not included in the project.

Options

You can use the following options with the FSRUNPRJ macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

LEAD= *n*
specifies the number of periods into the future in which multi-step forecasts are made. The default value is 12.

RETAINCHOOSE= YES | TRUE | NO | FALSE
clears any user–specified model selections and resets the series to the model automatically selected by SAS Forecast Studio. The default value is **`YES`**.

Results

The FSRUNPRJ global macro indicates whether the FSRUNPRJ macro finishes successfully or encounters errors: &FSRUNPRJ=SUCCESS |WARNING| ERROR.

Note: A warning message displays if there are override conflicts or reconciliation failures.

A series of global macro variables are populated after a SAS Forecast Server project is loaded. These macro variables describe the project. Using these macro variables, you can write SAS code that is generic enough to use across multiple projects. For example, you can use this code to explore and use data in your SAS Forecast Server projects. You can also use these macro variables in SAS Stored Processes. For more information, see "What Is an Internal Report Parameter?" on page 83.

Example

```
%fsrunprj(projectname=pd1,
user=sasuser,
password=saspass,
host=localhost:6411,
method=FORECAST,
importdata=NO,
retainchoose=YES
);
```

FSSETOWN Macro

The FSSETOWN macro assigns an owner to a SAS Forecast Server project.

Note: Only an administrative user can run this macro.

Syntax

%FSSETOWN (PROJECT=, OWNER=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSSETOWN macro. The required arguments are separated by commas.

PROJECT= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

OWNER= *user-name*
specifies the user name that will own the SAS Forecast Server project.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSSETOWN macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **`default`**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **Default**.

Results

The FSSETOWN global macro indicates whether the FSSETOWN macro finishes successfully or encounters errors: &FSSETOWN=SUCCESS| ERROR.

Example

```
%fssetown(project=Project3,
owner=sastrust,
user=sasadm,
password=sasadmpass,
environment=Default
);
```

FSSETPUB Macro

The FSSETPUB macro enables public access to a SAS Forecast Server project.

Note: If the FSSETPUB macro is used on a project created with an older version of SAS Forecast Server, then the project is converted to the current version.

Syntax

%FSSETPUB (PROJECT=, ISPUBLIC=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSSETPUB macro. The required arguments are separated by commas.

PROJECT= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

ISPUBLIC= TRUE | FALSE | YES | NO
specifies whether the project should be available to all users.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the password that you use to log on to SAS Forecast Studio.

Options

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

Results

The FSSETPUB global macro indicates whether the FSSETPUB macro finishes successfully or encounters errors: &FSSETPUB=SUCCESS| ERROR.

Example

```
%fssetpub(project=Project3,
ispublic=YES,
user=sasadm,
password=sasadmpass,
environment=Default
);
```

FSUNREG Macro

The FSUNREG macro unregisters an existing SAS Forecast Server project from the metadata server. The project files will not be deleted from the disk. If needed, you can register the project again from the project files.

Note: You must have administrative capabilities to run this macro variable.

Syntax

%FSUNREG (USER=,PROJECTNAME=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSUNREG macro. The required arguments are separated by commas.

USER= *username*
specifies the user name that you use to log on to SAS Forecast Studio.

PROJECTNAME= *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSREGPRJ macro. Options must follow the required arguments and are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **`Default`**.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

Results

The FSUNREG global macro indicates whether the FSUNREG macro finishes successfully or encounters errors: &FSUNREG=SUCCESS | ERROR.

Example

```
%fsunreg(projectname=prdat2,
user=sasadm,
password=sasadmpass
);
```

FSUNRENV Macro

The FSUNRENV macro unregisters a SAS Forecast Server environment. Unregistering an environment removes the existing SAS Forecast Server projects and environment from the metadata server without deleting the project files from the disk. You can use the FSREGENV macro variable to create an environment and register its projects.

Syntax

%FSUNRENV (ENVIRONMENT=, USER=, PASSWORD= [, options]) ;

Details

Required Arguments

The following arguments must be used with the FSUNRENV macro. The required arguments are separated by commas.

ENVIRONMENT= *environment-name*
specifies the name of the SAS Forecast Server environment. The default environment is **Default**.

USER= *user-name*
specifies the user name that you use to log on to SAS Forecast Studio.

PASSWORD= *password*
specifies the user password that you use to log on to SAS Forecast Studio.

Options

You can use the following options with the FSUNRENV macro. Options must follow the required arguments and are separated by commas.

SASENVIRONMENT=*environment-name*
specifies the symbolic name of the SAS environment where the middle tier for SAS Forecast Server is saved. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is **default**.

Results

The FSUNRENV global macro indicates whether the FSUNRENV macro finishes successfully or encounters errors: &FSUNRENV=SUCCESS | ERROR.

Example

```
%fsunrenv(environment=Default,
user=sasadm,
password=Password1,
sasenvironment=default
);
```

Examples

Example 1: Migrating SAS Forecast Server 2.1 Projects to SAS Forecast Server 4.1

FSEXPALL, FSIMPALL, FSMIGALL, and FSMIGPRJ Macros

This example explains how to batch update projects created with SAS Forecast Server 2.1 to a SAS Forecast Server 4.1 middle tier host installation on a separate machine. This example assumes the following:

- Both middle tier hosts are running Microsoft Windows.
- All SAS products are installed in their default installation folders.
- The name of the machine hosting the SAS Forecast Server 2.1 middle tier is fs21.
- The name of the machine hosting the SAS Forecast Server 4.1 middle tier is fs41.
- The project archive folder on fs21 is **`C:\SAS\ForecastStudio\Archives`**.
- The project archive folder is accessible on fs41 using the **`\\fs21\SAS\ForecastStudio\Archives`** path.
- The SAS administrator user ID is fsadm on both machines.

To migrate the SAS Forecast Server 2.1 projects:

1. If you have not done so, install the SAS Forecast Server macros on the SAS Forecast Server 2.1 middle tier. The macros are automatically installed on the SAS Forecast Server 4.1 middle tier. For more information about installing these macros for version 2.1, see the *SAS Forecast Server 2.1 Administrator's Guide*.
2. Archive the projects on fs21:
   a. Open SAS in Display Manager mode on fs21.
   b. Define the library where the OUT= data set is written. Both hosts must have access to the library. This example uses the Archive folder on fs21.

```
libname migrate 'C:\SAS\ForecastStudio\Archives\';
```

c. Submit the following SAS code to execute the FSEXPALL macro and export (archive) all projects:

```
%fsexpall(out=migrate.projects,
user=fsadm,
password=admpass,
namesuffix=_fs_export21
);
```

TIP Use the NAMESUFFIX= parameter to make the archive names unique. If an archive with the same name exists, then the archiving process fails for that project.

d. To verify that the projects were archived successfully, open the migrate.projects data set and review the ARCHIVEFAIL variable. ARCHIVEFAIL equals 0 if the project was successfully archived. For more information about the global macro variables for the FSEXPALL macro, see "FSEXPALL Macro" on page 134.

3. Import all projects to fs41:

a. Open SAS in Display Manager mode on fs41.

b. Define the library that contains the PROJECTDS= data set.

```
libname migrate '\\fs21\SAS\ForecastStudio\Archives\';
```

c. Run the FSIMPALL macro to import all projects. This macro creates the project data folders and registers the projects with the SAS Metadata Server.

```
%fsimpall(projectds=migrate.projects,
archivefolder=\\fs21\SAS\ForecastStudio\Archives,
user=fsadm,
password=admpass,
environment=Default
);
```

4. Migrate all projects to the current version using the FSMIGALL macro:

```
%fsmigall(out=work.migration,
user=fsadm,
password=admpass
);
```

TIP As an alternative to FSMIGALL, you can use the FSMIGPRJ macro to migrate single projects. For more information, see "FSMIGPRJ Macro" on page 145. If a project has not been migrated, then SAS Forecast Studio prompts you to do so the first time the project is opened.

TIP You can also migrate single projects or migrate projects in batch using the SAS Forecast Server Plug-in for SAS Management Console.

5. To verify that all of the projects were successfully migrated, open the work.migration data set and review the ISMIGRATED variable. ISMIGRATED equals 1 if the project was successfully migrated.

Example 2: In-Place Migration of SAS Forecast Server 2.1 Projects to SAS Forecast Server 4.1

FSMIGALL Macro

If you have upgraded the SAS Forecast Server middle tier on the current machine to SAS Forecast Server 4.1, you can use the FSMIGALL macro to migrate all of the projects registered in the SAS Metadata Server.

Note: Before following this example, you must create an environment and configure it to point to the old project location. You can use the SAS Forecast Server Plug-in for SAS Management Console to create and configure the environment. For more information, see the Help for the SAS Forecast Server Plug-in for SAS Management Console in SAS Management Console.

Alternatively, you can use the FSMIGPRJ macro to migrate single projects.For more information, see "FSMIGPRJ Macro" on page 145. If a project has not been migrated, then SAS Forecast Studio prompts you to do so the first time the project is opened.

Note: You can also migrate single projects or migrate projects in batch using the SAS Forecast Server Plug-in for SAS Management Console.

To migrate all projects using FSMIGALL:

1. Open SAS in Display Manager mode on the SAS server machine where the macros are installed.
2. Submit the following SAS code:

```
%fsmigall(out=work.migration,
user=fsadm,
password=admpass,
environment=Default,
);
```

3. To verify that all of the projects were successfully migrated, open the work.migration data set and review the ISMIGRATED variable. ISMIGRATED equals 1 if the project was successfully migrated.

For more information, see "FSMIGALL Macro" on page 144.

Example 3: Create a SAS Forecast Server Project

FSCREATE Macro

In this example, the FSCREATE macro is used to create a SAS Forecast Server project.

The SASHELP.PRICEDATA data set is used to create a hierarchical project. The hierarchy of the project is defined by the following variables:

- regionName
- productLine
- productName

This example also uses the following variables to create the project:

- date is the time ID variable.
- sale is used to model the total sale as a function of the price variable and the discount variable.
- price1-price3 are used as the reporting variables.

The project is reconciled in a middle-out fashion starting from the productLine level, and the disaggregation method is "equal split of the difference".

You can accept the default values for all other project creation options.

Note: You can use the sasdemo user ID to create the project. However, the project must be publicly available so that other users can open it and modify it.

To create a project using the FSCREATE macro:

1. Open SAS in Display Manager mode on the SAS server machine where the macros are installed.
2. Run the macro for creating a project by submitting the following SAS code:

```
%fscreate(projectname=mypricedataprj,
user=sasdemo,
password=Password1,
environment=Default,
data=sashelp.pricedata,
id=date,
by=regionName productLine productName,
hierarchy=YES,
var=sale,
input=price discount,
reporting=price1 price2 price3,
aggregate=NONE total(sale),
disaggregation=EQUALSPLIT,
reconciliation=BOTTOMUP,
publicaccess=YES
);
```

For more information, see "FSCREATE Macro" on page 116.

Example 4: Archive a SAS Forecast Server Project

FSEXPORT Macro

This example exports a SAS Forecast Server project to an archive file using the FSEXPORT macro.

This example assumes the following:

- pd1 is the project's name.
- ArchPD1 is the archive's name.
- Project pd1 is the archive's description.
- There is only one SAS Forecast Server environment.
- The middle tier server is your local machine so the ENVIRONMENT= and HOST= parameters are not specified.

To archive a project using the FSEXPORT macro:

1. Open SAS in Display Manager mode on a SAS server machine.
2. Submit the following SAS code:

```
%fsexport(projectname=pd1,
archivename=ArchPD1,
description=Project pd1,
user=sasuser,
password=saspass,
archivefolder=C:\SAS\ForecastServer\Archives,
environment=Default
);
```

For more information, see "FSEXPORT Macro" on page 135.

Chapter 13

Troubleshooting SAS Forecast Server

Gathering Information

Overview

When you are troubleshooting unexpected application behavior, it is important to isolate and describe the problem and the context in which it occurs. The following are the general classes of information that can expedite resolution of a technical problem:

- operating system environmental and configuration information

- detailed problem description
- log files
- other files or screen shots
- sample test data

Use the following table to help gather information. Providing this information helps SAS Technical Support reproduce and fix your problem.

Table 13.1 *Information Gathering Checklist*

Task	Done
Details of your operating environment.	[ ]
Detailed description of the problem (including what it takes to reproduce it).	[ ]
Sample data that would help to reproduce the problem.	[ ]
Obtain log files.	[ ]
Full Java stack trace from the error page.	[ ]

Operating System Environmental and Configuration Information

If you request help from SAS Technical Support, then providing the following information about your installation can result in faster resolution of the problem:

- hardware platform, operating environment (including SAS version number), amount of physical memory, and number of processors
- JDK version
- JRE version
- the SAS Forecast Server version number and patch level
- the configuration settings for SAS Forecast Server. You can get this information from the **Settings** and **Advanced** tabs in the Configuration Manager in SAS Management Console.
- server language and locale

Note: You must provide the preceding information only once, unless it has changed from previous reports.

Problem Description

Provide a scenario description that includes as much information as possible. Include a description of the general task that you are trying to accomplish, your role and permissions, and what has happened during the SAS session. Provide details such as the following:

- Are you working with new data or updating existing data?

- How easy is the problem to reproduce?
- What browser and version are you using?
- Is the problem locale-specific? If so, which locales are having problems?

Sample Test Data

If possible, capture the information entered that caused the problem. In certain situations, SAS Technical Support might request your data load files so that they can better replicate your operating environment.

Log Files

SAS Forecast Studio Log

To view the log in SAS Forecast Studio, select **Tools** ⇨ **SAS Log**.

To include additional information in the log:

1. Start SAS Management Console and connect as a SAS administrator.
2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
4. Click the **Settings** tab.
5. In the selection pane, click **SAS Forecast Server**.
6. Set these options:
   - Disable filtering. By default, messages about internal activities are omitted from the SAS log.
   - Enable the inclusion of JDBC messages. By default, JDBC messages are not included in the SAS log.

   Click **OK**.

Java Stack Traces

When sending information to SAS Technical Support, send the complete full text of the Java stack trace. The logs for SAS Forecast Server are in the standard location for the application container. An example of this path is ***SAS_CONFIG*\Web\Logs**. Log files for SAS Forecast Server will be in the form **SASForecastServer*.log**. SAS Technical Support prefers that you do not send a screen shot of this information because often the screen shots do not include the full text of the trace.

If possible, also include the log for your Web Application Server. For JBoss, these logs are available from **\jboss-as\4.3.0\server\SASServer1\log**.

Additional Resources

The Status Page for SAS Forecast Server

From the Status page for SAS Forecast Server, you can validate the status of your server, confirm your licensed product, and determine whether your product versions are compatible. This page also provides details about current sessions including the resources that are being used.

JMX Beans

Although the Status page for SAS Forecast Server provides a summary of the run-time state of the middle tier, you cannot modify the state of the middle tier from this page. The SAS Forecast Server provides a simple JMX bean that you can use to modify the state. The JMX bean provides some details about the run-time state, but these details are directly related to the actions that are supported by the JMX bean.

These actions are supported by the JMX bean:

- General operation:
  - **`isInitialized()`** — reports whether the middle tier has completed the initialization that is performed at start up
  - **`reloadConfiguration()`** — reloads the product configuration from the metadata and attempts to apply any changes to the running application.

    Note: Configuration information is also cached in the Configuration Service. This service must be reloaded prior to the **`reloadConfiguration()`** call in order to avoid getting the configuration details from an old cache.
- Server validation:
  - **`getServerStatus()`** — returns the validation information for all tested SAS Workspace Servers.
  - **`refreshServer(`*`server-name`*`)`** — clears the stored validation information for the specified SAS Workspace Server. Clearing this information results in the server being retested when the server is next accessed.
  - **`refreshAllServers()`** — clears the stored validation information for all SAS Workspace Servers.
- Session management
  - **`getSessions()`** — reports all active sessions in the middle tier.
  - **`killSession(`*`ID`*`)`** — forces the specified session to close.

    Note: Only use this action to clean up sessions that no longer have clients associated with them. For example, you might use this action when a client process must be manually terminated.

TIP After executing a JMX call, you can confirm your change by using the Status page for SAS Forecast Server.

Project Owner Cannot Access Project

If you cannot access a SAS Forecast Server project that you should own, you might want to check the owner information that is stored with the project. SAS Forecast Server stores the identity of the owner as a token value that is derived from the user ID. However, the token value might not be identical to the user ID.

To determine the owner of the project, see these locations in SAS Forecast Studio:

- the Project Properties dialog box
- the **Owner** column of the table in the Projects dialog box

You can also determine the owner of the project by using the SAS Forecast Server plug-in for SAS Management Console and using the SAS macros.

To view the token value for the user who is currently logged in, see these locations:

- the active sessions table on the SAS Forecast Server Status page
- the **Show my (*identity-token*) projects only** check box in the Projects dialog box

Note: In some cases, a token mismatch can result when migrating projects from a previous release. Prior to the 4.1 release, the user ID at the logon prompt was used to generate these tokens. Starting in SAS Forecast Server 4.1, the name attribute of the metadata identity is used instead. In cases where the user ID and metadata name differ, the values of the project owner must be updated during the migration process.

Troubleshooting the Java Web Start Client

Java Version Missing for Java Web Start

Java Web Start does not work because the required version of JRE is not installed on the client tier. For SAS 9.3 products, the required Java version 1.6 or later. To download the JRE, see `http://support.sas.com/resources/thirdpartysupport/v93`.

JNLP File Is Not Signed

When you launch the Java Web Start client, you see a warning that the JNLP file is not signed. For SAS Forecast Server, the JNLP files are dynamically generated and therefore cannot be signed. You can ignore this warning.

Enable the JWS Cache

The Java Web Start client does not work properly if the Java Web Start cache is disabled. By default, the cache is enabled. If this caching is disabled for any reason, complete these steps to turn caching on:

1. On Windows 7, select **Start** ⇨ **Settings** ⇨ **Control Panel** ⇨ **Programs** ⇨ **Java**.
2. Click the **General** tab.

3. In the **Temporary Internet Files** section, click **Settings** and select the **Keep temporary files on my computer** check box.

Cannot Select a SAS Environment from the Log On Dialog Box

During the logon process, SAS Forecast Server looks for a URL that references a sas-environment.xml file. This file contains the list of SAS environments that a user can select from during log on. SAS Forecast Server searches these locations in order of precedence:

1. the **`sas.env.definition.location`** (a Java system property).
2. the **`env.definition.location`** (a Java system property).
3. the SAS_ENV_DEFINTION_LOCATION environment variable for the operating system.
4. the configured URL that is stored in the SASENVIRONMENTSURL property in **`%{sas.home}/sassw.config`**. (**`sas.home`** is a Java system property.)
5. the configured URL that is stored in the SASENVIRONMENTSURL property in **`$SASHOME/sassw.config`**. (**`SASHOME`** is an environment variable for the operating system.) During the deployment of SAS Forecast Studio, you are prompted for the value to store in the sassw.config file. If the SAS Forecast Server client cannot locate this URL, verify that the information in the sassw.config file is correct.

Launching SAS Forecast Studio using Java Web Start does not require this URL to be configured. However, if the URL has been configured on either the client machine or in the configuration properties for Java Web Start, then SAS Forecast Server might generate an error. To resolve this error, look in the previous locations and verify that the valid URL is specified.

SAS Forecast Server Cannot Locate Any Compatible SAS Environments at Logon Time

During the logon process, SAS Forecast Server runs a compatibility test on each SAS environment. Primarily this test is used to ensure that each of the SAS environments that a user can select from the Log On dialog box are associated with a compatible deployment of SAS Forecast Server. Only SAS environments that pass this test appear in the Log On dialog box.

If no SAS environments pass this test, an error message states that no SAS environments could be found. If you see this message, verify the following requirements:

- The sas-environment.xml file that you are using includes at least one SAS environment with a SAS Forecast Server deployment.
- The version of the SAS Forecast Server deployment matches the version of the client.
- The servers associated with the deployment are running and responsive.

Troubleshooting the SAS Forecast Server Plug-in for SAS Management Console

Cannot Access Application Plug-ins in SAS Management Console

The SAS Forecast Server Plug-in for SAS Management Console might not be available in these situations:

- When you log on to SAS Management Console, the SAS Forecast Server Plug-in for SAS Management Console does not display under the **Application Management** node. To resolve this issue, assign the user to the **Management Console: Advanced** role. This role is required so that the SAS Forecast Server Plug-in for SAS Management Console and other solution-specific plug-ins are available for that user.
- When you try to access the SAS Forecast Server Plug-in for SAS Management Console, your logon information is rejected. The plug-in uses your logon credentials for SAS Management Console. Therefore, this logon identity must be configured as a product user and have the appropriate capabilities assigned to it.

Note: Do not configure internal accounts (such as sasadm@saspw) for use with SAS Forecast Server in order to solve this problem. You should log in to SAS Management Console with a product user account.

User Is Unrestricted

When using the SAS Forecast Server plug-in, you might see a warning that the current user is unrestricted. The unrestricted designation is used for special accounts, such as sasadm@saspw. This designation also effectively disables all of the metadata permissions and capabilities. So in a sense, an unrestricted account is assigned all of the permissions and capabilities that are available for the product. Using an unrestricted user is generally not recommended because the unrestricted designation applies only to the SAS Metadata Server and not to other resources, such as the SAS Workspace Server. So unless the user has been configured to use the product, then any actions taken by the unrestricted user will fail and could result in corrupt data. Therefore, the unrestricted designation should be used only with special accounts (such as sasadm@saspw) that are not used with the product.

Environment Action Fails

When you perform maintenance actions on environments (copy, move, delete, and so on), exclusive access to the environment is typically required for the duration of the action. During this time, the environment and all projects within it are inaccessible to users. Similarly, if the environment or one of its projects is already in use when the action is attempted, then the action fails. Therefore, it is recommended that you coordinate with users and establish times during which you can perform environment maintenance.

Troubleshooting the SAS Forecast Server Macros

Classpath Variable Is Not Set

You do not have to set the classpath variable before using the macros. If you see the following note while executing the macros, you can safely ignore it.

```
NOTE: Could not initialize classpath. Classpath variable is not
set.
```

log4j System Property Is Not Initialized

You do not need to configure the log4j logging service for Java. If you see the following warning while executing the macros, you can safely ignore it.

```
log4j:WARN No appenders could be found for logger java—class—name
log4j:WARN Please initialize the log4j system properly.
```

Performance Tuning for SAS Forecast Server

For scalability issues, slow system performance, or failures caused by memory settings, you can change the default settings for SAS Forecast Studio and Java Web Start in the following tiers:

- SAS Forecast Studio client tier
- SAS server tier

To improve performance:

1. Change the maximum heap size. By default, this value is `512m`. For information about formatting and value constraints, see the JRE documentation.

   You can change this value in either of these ways:

   - Change the `Xmx` value in the .ini files.
   - by setting the **Maximum heap size** property in SAS Management Console.
     1. Start SAS Management Console and connect as a SAS administrator.
     2. Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
     3. Right-click the **Forecast Server 4.1** node and select **Properties**. The Forecast Server 4.1 Properties dialog box appears.
     4. Click the **Settings** tab.
     5. In the selection pane, click **Virtual Machine Options**.
     6. Specify a value for the maximum heap size. Click **OK**.

2. Increase the JVM memory for the server, middle tier, and client by specifying the **-Xmx1g** command.

The Windows configuration files are at the following locations:

- Client (SAS Forecast Studio): **SAS_HOME\SASForecastStudio\4.1\forecaststdo.ini**
- JRE variables **SAS_HOME\wrapper.conf**

For more information about changing your system settings to improve system performance, see the section about best practices for configuring the middle tier in the *SAS Intelligence Platform: Middle-Tier Administration Guide* at **http://support.sas.com/documentation/onlinedoc**.

Glossary

client application
an application that runs on a client machine.

client tier
the portion of a distributed application that requests services from the server tier. The client tier typically uses a small amount of disk space, includes a graphical user interface, and is relatively easy to develop and maintain.

data set
See SAS data set.

descriptor information
information about the contents and attributes of a SAS data set. For example, the descriptor information includes the data types and lengths of the variables, as well as which engine was used to create the data. SAS creates and maintains descriptor information within every SAS data set.

environment
a virtual container of run-time settings for SAS Forecast Server client sessions.

foundation services
See SAS Foundation Services.

Integrated Object Model server
See IOM server.

IOM server
a SAS object server that is launched in order to fulfill client requests for IOM services. Short form: IOM server.

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.

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.

object 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).

SAS 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 data files contain data values in addition to descriptor information that is associated with the data. SAS data views contain only the descriptor information plus other information that is required for retrieving data values from other SAS data sets or from files whose contents are in other software vendors' file formats.

SAS 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, activity logging, event management, information publishing, and stored process execution.

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

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 Workspace Server
a SAS IOM server that is launched in order to fulfill client requests for IOM workspaces.

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.

service
one or more application components that an authorized user or application can call at any time to provide results that conform to a published specification. For example, network services transmit data or provide conversion of data in a network, database services provide for the storage and retrieval of data in a database, and Web services interact with each other on the World Wide Web.

spawner
See object spawner.

Index

I

J

L

M

O

P

R

S

T

U

W