THE POWER TO KNOW.

SAS® Forecasting for Desktop 12.1

Administrator's Guide, Second Edition

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2012. *SAS® Forecasting for Desktop 12.1: Administrator's Guide, Second Edition*. Cary, NC: SAS Institute Inc.

SAS® Forecasting for Desktop 12.1: Administrator's Guide, Second Edition

ISBN (electronic book)

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

Printing 1, October 2012

Contents

Using This Book

Audience

This book is for administrators who need to install, configure, and optimize SAS Forecasting for Desktop. SAS and other programming expertise is not required.

Prerequisites

To ensure that your system meets the appropriate requirements, review the system requirements documentation before you install SAS Forecasting for Desktop. For more information, see `http://support.sas.com/documentation/installcenter/en/ikforecastwofrsr/65720/HTML/default/index.html`.

Accessibility

Accessibility and Compatibility Features

SAS Forecasting for Desktop has been tested with assistive technology tools. It 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 (2008 draft proposal initiative update). For more information about the accessibility of the SAS Forecasting for Desktop, see the *SAS Forecast Studio: User's Guide*.

For detailed information about the accessibility of this product, send e-mail to **accessibility@sas.com** or call SAS Technical Support.

Recommended Reading

- the online Help for SAS Forecast Studio
- *SAS Forecast Studio: User's Guide*

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

Part 1

Overview of the SAS Forecasting for Desktop

1

Understanding the SAS Forecasting for Desktop

What Is SAS Forecasting for Desktop?

SAS Forecasting for Desktop is a forecasting suite that enables you to quickly run smaller scale forecasting tasks, bringing automatic forecasting to the small and midsize market. Utilizing the same underlying technology as SAS Forecast Server, SAS Forecasting for Desktop generates statistically based forecasts without the need for human intervention, unless so desired.

The suite consists of desktop versions of many of the client applications from the SAS Forecast Server product suite:

- SAS Forecast Studio for Desktop
- SAS Forecast Project Manager for Desktop
- SAS Forecast Batch Interface

The primary difference among these suites is that SAS Forecast Server includes the required SAS Forecast Server middle-tier component. The SAS Forecasting for Desktop suite embeds this middle-tier logic directly into each client. SAS Forecasting for Desktop clients also use alternative designs in some components to avoid specific server dependencies.

SAS Forecasting for Desktop uses many procedures and options from other SAS products, such as SAS/ETS, Base SAS, and SAS/GRAPH. For more information about how SAS Forecasting for Desktop relates to other SAS products, see *SAS Forecast Studio: User’s Guide*.

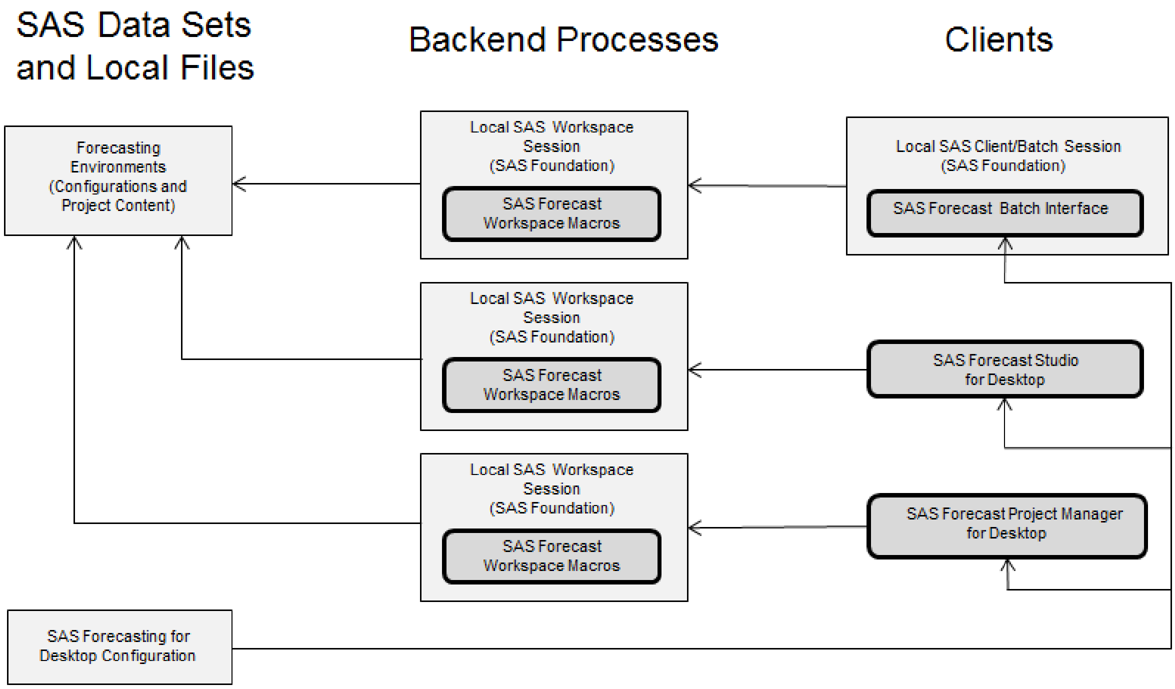
The user interface for SAS Forecast Studio for Desktop is virtually identical to the user interface for SAS Forecast Studio, so you can use the *SAS Forecast Studio: User's Guide* to work with this product. Online Help is also available within SAS Forecast Studio for Desktop. (This help is shared with SAS Forecast Studio.)

Overview of the SAS Forecasting for Desktop Architecture

Architecture Diagram

The following figure shows the SAS Forecasting for Desktop architecture.

Figure 1.1 The SAS Forecasting for Desktop Architecture


SAS Data Sets and Local Files
Backend Processes
Clients
Forecasting Environments (Configurations and Project Content)
Local SAS Workspace Session (SAS Foundation)
SAS Forecast Workspace Macros
Local SAS Client/Batch Session (SAS Foundation)
SAS Forecast Batch Interface
Local SAS Workspace Session (SAS Foundation)
SAS Forecast Workspace Macros
SAS Forecast Studio for Desktop
Local SAS Workspace Session (SAS Foundation)
SAS Forecast Workspace Macros
SAS Forecast Project Manager for Desktop
SAS Forecasting for Desktop Configuration

Components of SAS Forecasting for Desktop

- SAS Forecast Studio for Desktop, which is the graphical interface to the forecasting and time series analysis procedures that are contained in SAS High-Performance Forecasting and SAS/ETS software
- SAS Forecast Project Manager for Desktop, which is an administrative component that enables you to manage projects, environments, and libraries
- SAS Forecast Batch Interface, which enables you to use SAS macros to manage projects and environments. The SAS Forecasting for Desktop license enables use of all of the macros except for FSCREATE.

  Note: The macros referenced here and throughout this document refer to the SAS Forecast Batch Interface package only unless explicitly stated otherwise.

Concepts Used in SAS Forecasting for Desktop

Users
: The people who use SAS Forecasting for Desktop on your installation.

Forecasting environments
: The product workspaces that are used for product sessions. They are used by SAS Forecasting for Desktop to help users manage their project workspaces within the product.

Note: Forecasting environments are not the same as SAS environments. SAS environments are used during logon actions in server-based deployments to locate the available SAS deployments.

Libraries
: The standard means of storing and managing data sets in SAS. Libraries can be stored within the environment or externally. Internally stored libraries (called automatic environment libraries) can be used as scratch areas in which data can be stored across multiple sessions. Externally stored libraries (configured environment libraries) are more flexible but must be managed manually. Libraries defined through the SAS Foundation can also be used.

Projects
: The standard means of storing the files and other states related to a particular forecasting problem. SAS Forecasting for Desktop supports both hierarchal forecasting with a single dependent variable and non-hierarchal forecasting with multiple dependent variables.

SAS Forecast Server: Interplay and Differences

- SAS Forecasting for Desktop does not include a SAS High-Performance Forecasting license while SAS Forecast Server does.
- HPF procedures can be run by the SAS Forecasting for Desktop clients. However, users cannot submit code directly that uses the procedures due to the license.
- Projects can be transferred to and from SAS Forecast Server deployments.
- Only projects with 1000 or fewer time series can be used under the SAS Forecasting for Desktop license. Actions that would violate this limit will be blocked by the software. The SAS Forecast Server license does not include this limitation.
- Environments cannot be shared or transferred between SAS Forecasting for Desktop and SAS Forecast Server.
- SAS Forecasting for Desktop uses different %FSLOGIN arguments when creating product sessions for the SAS Forecast Batch Interface macros.
- On SAS Forecasting for Desktop, you do not specify a host server when defining environments. The host is understood to be the local machine.
- The %FSCREATE macro in the SAS Forecast Batch Interface cannot be used under the SAS Forecasting for Desktop license. The SAS Forecast Server license does not include this.
- SAS Stored Processes cannot be run from the SAS Forecast Studio for Desktop while they can be run from SAS Forecast Studio in SAS Forecast Server.

Part 2

Installing the SAS Forecasting for Desktop

2

Pre-installation Tasks

SAS Environment URL Prompt

In the SAS Forecasting for Desktop, you can set the location of the SAS environment URL to enable SAS Forecast Batch Interface macro access to SAS Forecast Server deployments. This type of prompt is normally associated only with networked deployments. It appears here due to the inclusion of SAS Forecast Batch Interface. During deployment, you are prompted by the SAS Deployment Wizard to specify a URL location of a SAS environment file. SAS environment files are used by clients, in server-based SAS deployments, to locate the available SAS deployments. This URL is not required when you use the SAS Forecast Batch Interface with the SAS Forecasting for Desktop deployment. If you intend to connect to SAS Forecast Server deployments by using the local macros, you might want to determine the location to use for the SAS Environment URL as a pre-installation task.

If you decide to configure this URL, see “Configuring the SAS Environment File” in the *SAS Intelligence Platform: Middle-Tier Administration Guide* for more information about the structure of the environment file, located at `http://support.sas.com/documentation/onlinedoc/intellplatform`.

3

Installation Tasks

About the Deployment of SAS Forecasting for Desktop

When you deploy SAS Forecasting for Desktop, you deploy all of the components that are part of the SAS Forecasting for Desktop architecture. For more information, see "What Is SAS Forecasting for Desktop?" on page 3.

How to Install and Configure SAS Forecasting for Desktop

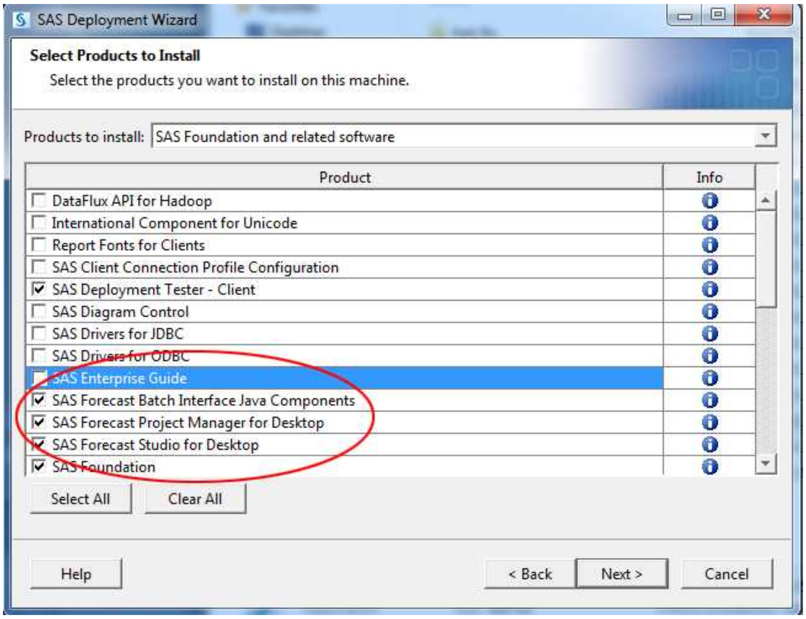
To install and configure SAS Forecasting for Desktop, you use the SAS Deployment Wizard and follow the basic process described in the *SAS Intelligence Platform: Installation and Configuration Guide*. Some steps in the SAS Deployment Wizard are specific to installing and configuring SAS Forecasting for Desktop. Here is the specific information that you need to install SAS Forecasting for Desktop:

1. Start the SAS Deployment Wizard from your SAS Software Depot. For example, on a Windows system, double-click the setup.exe file that is located in your SAS Software Depot folder.

2. In the Select Deployment Step and Products to Install step, verify that the following products will be installed:

   - SAS Forecast Batch Interface Java Components
   - SAS Forecast Project Manager for Desktop
   - SAS Forecast Studio for Desktop

   Note: Depending on what you license at your site, additional products might be installed. This list is specific to SAS Forecasting for Desktop.



Select Products to Install

Select the products you want to install on this machine.

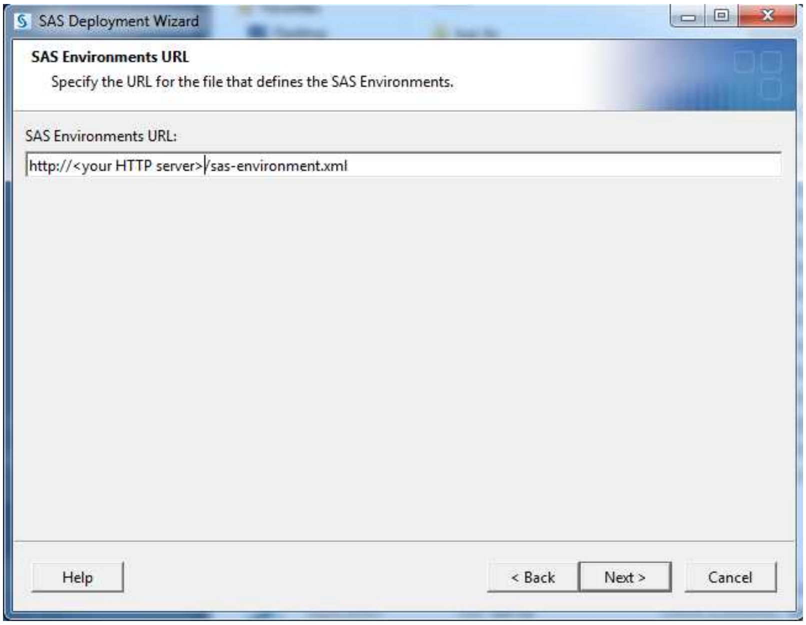
Products to install: SAS Foundation and related software

Product	Info
☐ DataFlux API for Hadoop	ⓘ
☐ International Component for Unicode	ⓘ
☐ Report Fonts for Clients	ⓘ
☐ SAS Client Connection Profile Configuration	ⓘ
☑ SAS Deployment Tester - Client	ⓘ
☐ SAS Diagram Control	ⓘ
☐ SAS Drivers for JDBC	ⓘ
☐ SAS Drivers for ODBC	ⓘ
☐ SAS Enterprise Guide	ⓘ
☑ SAS Forecast Batch Interface Java Components	ⓘ
☑ SAS Forecast Project Manager for Desktop	ⓘ
☑ SAS Forecast Studio for Desktop	ⓘ
☑ SAS Foundation	ⓘ

Click **Next**.

3 If you intend to connect to SAS Forecast Server deployments using the SAS Forecasting Batch Interface macros from this installation, then specify a URL location of the SAS environment file in the **SAS Environments URL** field. For more information, see "SAS Environment URL Prompt" on page 9.

Note: Leave this field blank if you do not intend to connect to SAS Forecast Server deployments using the SAS Forecasting Batch Interface macros from this installation. The URL can be set after installation if your needs change.



SAS Environments URL

Specify the URL for the file that defines the SAS Environments.

SAS Environments URL:

http://<your HTTP server>/sas-environment.xml

4 In the Deployment Summary, review the list of products that you are about to install, and click **Start**.

5 In the Additional Resources step, review the additional resources and complete the manual configuration instructions in the Instructions.html file.

Click **Finish** to exit the SAS Deployment Wizard.

4

Post-installation Tasks

Post-Installation Overview

Customize the Configuration Paths

Customizations that are made after the installation of the SAS Forecasting for Desktop apply only to the current user by default.

The SAS Forecasting for Desktop suite operates by establishing a configuration area for the local deployment. By default, the suite creates a separate configuration for each user. That is, the out-of-the-box behavior is customized for personal use, so it isolates the configurations of different users.

However, the suite also supports the use of shared configuration areas. In this case, a single configuration is used to support multiple users. In effect, this shared configuration is parallel to the behavior under the SAS Forecast Server suite. Due to centralizing the configuration and content, this alternative offers some advantages in terms of content sharing and reduced management effort.

Note: Each configuration area should be treated like a separate deployment in terms of management. For example, the same forecasting environment location should not be shared between two configurations. In order to share content areas in this way, the configuration areas must be merged into a single area.

You can set up shared configurations by specifying where the application configuration is maintained with VM options in the forecaststdw.ini file as shown below.

```
-Ddesktop.home=filepath
```

When sharing a configuration, the chosen path should be user-insensitive and must be accessible to all users. When this path is not manually specified, the configuration area is automatically created within the current user's home directory. This assumed default produces the isolated per-user configurations mentioned above. The home directory for the current user is reported automatically by the local operating system. However, you can also use a Java VM argument like the one above to customize this path . For a more detailed discussion of these paths and how to customize them, see“Understanding the Desktop Configuration Files” on page 20.

Other Customizable Areas

The number of post-installation customization tasks that you need to complete depends on your site. The product assumes default behavior and settings that should enable you to launch it out-of-the-box with no additional configuration in most situations. However, you can opt to customize the product behavior if some defaults are undesirable. You can customize the SAS Forecasting for Desktop in the following areas:

- specify a specific forecasting environment. For more information, see “Creating and Configuring a Forecasting Environment” on page 14.
- set the permissions appropriately for each forecasting environment file area. For more information, see “Set File System Permissions” on page 15.
- specify the identity details and capabilities for each user of the SAS Forecasting for Desktop. For more information, see “Configuring Ownership Status of a Project” on page 22.

Creating and Configuring a Forecasting Environment

About Forecasting Environments

A forecasting environment is a product workspace for product sessions. Environments can be used to organize your projects and to control access to SAS Forecasting for Desktop projects.

If you have more than one environment, do not use the same directory for multiple environments. In general, one file system location should be used by a single environment across all deployments. This is also true when considering deployments of different types, such as SAS Forecasting for Desktop and SAS Forecast Server. 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.

When you launch SAS Forecast Studio for Desktop, a default environment is created automatically if no forecasting environment exists and the appropriate permissions are set. Therefore, you need to create an environment only if you want to either control the location of environment files or use multiple environments. You can create additional environments with the SAS Forecast Project Manager for Desktop.

Create a Forecasting Environment in SAS Forecast Project Manager for Desktop

Although the structures are similar and present similar functionality, environment areas are not interoperable between the SAS Forecasting for Desktop and SAS Forecast Server suites. Only the transfer of individual projects between suite deployments is supported. The transfer of environments is not supported.

To create an environment:

1. To open the SAS Forecast Project Manager for Desktop, select **Start** ▸ **All Programs** ▸ **SAS** ▸ **SAS Forecasting for Desktop** ▸ **SAS Forecast Project Manager for Desktop 12.1**.
2. In the navigation tree, select the **System** node.
3. On the **Environments** tab, click ✸. The Create Environment dialog box appears.
4. Enter a name for the new environment.
5. (Optional) Type a description for the environment.
6. Specify the location of the environment in the file system.

   Note: If you have more than one environment or multiple deployments, here are some restrictions to note:

   - Your environments should not share the same file system location as any other environment in any deployment of SAS Forecasting for Desktop or SAS Forecast Server.
   - You should similarly not configure one environment to use a subdirectory or parent directory of any already existing environment from any deployment.
7. Click **OK**.

Set File System Permissions

Whenever a new environment is created, you must set appropriate file system permissions to the associated file area for all of the SAS Forecasting for Desktop users. Specifically, users who can access the forecasting environments must have both Read and Write 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 of the forecasting environment.
2. Right-click on this directory, and then select **Properties**.
3. Select the **Security** tab.
4. Select the users for the current machine. Subsets of users can be selected instead for more fine-grained security.

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. This parent directory is configured with the appropriate file system permissions as a post-installation step. In this way, the child directories inherit the permissions from the parent directory, so you do not need to configure each new environment. The recommended choice for the parent directory is ***!DESKTOPHOME*** **\Environments** because this is where the default environment is created.

Managing User Capabilities

SAS Forecasting for Desktop uses the user's identity and assigned capabilities to manage which privileges are granted to users. At run time, the operating system identity is used to identify the user, and default values are assigned for each capability.

As a post-installation task, you can customize either of these aspects to meet your site’s specific requirements. For more information about capabilities, see “Configuring User Capabilities” on page 22, “Capabilities for the SAS Forecasting for Desktop Users” on page 23, and “Configuring Ownership Status of a Project” on page 22.

Part 3

Configuration Tasks

5

Configuring the SAS Forecasting for Desktop

Configuration File Locations

The following table shows the configuration file locations used by the SAS Forecasting for Desktop and their default paths. Use this table as a reference.

Table 5.1 *Configuration File Locations*

Directory	Description	Windows Path
!SAS_HOME This document uses *!SAS_HOME* to represent this directory.	SAS installation directory.	`C:\Program Files\SASHOME`
!USERHOME This document uses *!USERHOME* to represent this directory.	Home path of the user who launched the process. This path is reported by the operating system, and the exact location will vary based on the type of operating system and user account. This location contains user-specific customization files.	`C:\Users\`*`userid`*

Directory	Description	Windows Path
!DESKTOPHOME This document uses *!DESKTOPHOME* to represent this directory.	Home path for the desktop configuration and status files. This location is where the configuration details and current state for the desktop mode are maintained. If this value is not explicitly set, the value is initialized to ***!USERHOME*** **\ForecastStudioDesktop**.	**C:\Users\*userid*** **\ForecastStudioDesktop**.

Managing Configuration Files

Understanding the Desktop Configuration Files

The SAS Forecasting for Desktop uses two file system locations to manage the desktop configuration. One file location, the “desktop home” or ***!DESKTOPHOME***, stores common and default information and is shared by all users. The other file location, the “user home” or ***!USERHOME***, is specific to the invoking user and is used to store user-specific information that takes precedence over the common or default settings. SAS Forecasting for Desktop uses certain rules to determine which file locations are used as “user home” and “desktop home”. The rules are as follows:

- The "user home" location is determined by checking the **forecasting.user.home** and then the **user.home** Java property values, in that order. Java properties can be set in the ini file that is located in the *!SAS_HOME* directory that is specific to each of the SAS Forecasting for Desktop clients; for the SAS Forecast Batch Interface macros, the Java properties are defined in the JREOPTIONS element within the SAS Foundation sasv9.cfg file, located at **C:\Program Files\SASHome\SASFoundation \9.3\sasv9.cfg**. You can set these values manually using the forecaststdw.ini file of each client if necessary. The **user.home** is a standard VM argument and will always be defined.
- The "desktop home" location is determined by checking the **forecasting.desktop.home** and then the **desktop.home** Java property values, in that order. These properties can be set in the same locations mentioned previously. The first defined value is used. If neither are defined, then the value is assumed to be the same as "user home". However, in all cases, this discovered value is actually the base value for the desktop home, which is obtained by appending "\SASForecastStudio" to the end of the discovered path.

Note: A problem with Java on Windows sometimes causes the Java **user.home** argument to report the incorrect home directory of the current user. In this case, the **user.home** or**forecasting.user.home** argument might need to be explicitly set to produce the intended default behavior.

Customizing the Desktop Configuration File Location

You might opt to set these locations explicitly if any of the following conditions apply:

- The user home is incorrectly reported to the application by your local operating system. Note that changing the user home implicitly changes the desktop home unless the latter has itself been explicitly set.
- You want to store the content somewhere other than the default location
- You want to set up a shared configuration. In this case, *!DESKTOPHOME* should be set to a non-user-specific location.

Regardless of how these paths are defined, you must ensure that all users have Read access to ***!DESKTOPHOME*** (the root of the desktop configuration area) and read-write access to all files and directories under ***!DESKTOPHOME*\Environments**. This last requirement enables the default environment to be automatically created. If no forecasting environment exists when a SAS Forecasting for Desktop client is launched, then the default environment is automatically created using the file location ***!DESKTOPHOME*\Environments\Default**. So all users user must have read-write permissions to this path.

By selecting the desktop home path based on the user home path, the default rules should always satisfy this permissions requirement automatically. That is, users should always have full access to their home directories and the contents within.

Configuring Users

Understanding Project Ownership and User Capabilities

Two areas of the SAS Forecasting for Desktop can be configured to customize the user experience. These areas include the following:

- Ownership

  Each SAS Forecasting for Desktop project has an owner and a sharing (also called public access) setting. The owner of a project is the local administrator of that particular content and can perform any action on the project other than re-assigning ownership to another user. When a project is shared, all other users of the forecasting environment that contains the project are permitted to open and modify the project. However, destructive actions like deleting the project are not permitted under this guest access. Such actions can usually be performed only by the project owner. The notable exception is discussed in the next entry.

- Capabilities

  SAS Forecasting for Desktop provides a set of controls that represent individual privileges that each user might be granted. Generally, each control represents access to an individual product feature. However, that is not always the case. In particular, if a user is granted the special Administer Product capability, then that user is exempted from restrictions that result from ownership and sharing settings in projects. Such users can perform any action that is not restricted by other controls (such as other capabilities). In addition, product administrators are the only users who can change the ownership of a project.

Configuring Ownership Status of a Project

In the SAS Forecasting for Desktop, the creator of a new project is assigned ownership of the project by default. The ownership is determined by the use of an identity token that is stored with other project settings. This token is derived from the user ID at run time. The value of the token might match the user ID in some cases, though you should not assume that this is always the case.

You can determine the value of the identity token for a particular user by launching the SAS Forecast Studio for Desktop client and looking at the Projects dialog box, which shows the list of available projects. A check box in that dialog box applies a filter to show only projects owned by that user. The label of the check box shows the identity token value in parentheses. In addition, the table in that same dialog box includes an **Owner** column that shows the owner (identity token value) for each of the available projects.

By default, the user ID is determined from the Java VM argument **`user.name`**, which is automatically reported by the local operating system. The desktop clients then use this reported value for both the user ID and user display name in the client interfaces. In some cases, you might want to customize one or both of these values. For example, if you routinely move projects between the SAS Forecasting for Desktop deployment and a SAS Forecast Server deployment, you might want the identity token in the desktop environment to match the identity token in the SAS Forecast Server environment. Otherwise, you would have to reassign the project ownership after each project transfer between the deployments.

Each user can customize his or her user ID and user display name by adding the **`user.name`** and **`user.display.name`** properties to the file ***!USERHOME*** **`\fsuser.properties`** as follows:

```
user.name=sasdemo
user.display.name=SAS Demo User
```

Note: The value provided by this **`user.name`** property is assumed to be the user ID, not the identity token value. You cannot manually set an identity token value because the provided value might violate restrictions on its form. Instead, you must synchronize the identity tokens indirectly by ensuring that the user ID is the same in both deployments.

You can directly modify the sharing and ownership settings that are stored with each project by using the provided client interfaces. You can modify the sharing status of a project by using the SAS Forecast Studio for Desktop. A check box for **Allow other users to view and edit this project** is at the bottom of the Project Properties dialog box. You can also modify the owner and sharing settings of a project with the **Access Setting** action in the SAS Forecast Project Manager for Desktop. Similarly, in the SAS Forecast Batch Interface, you can use the FSSETOWN and FSSETPUB macros to change the owner and sharing settings of a project.

Note: Only users who are granted the Administer Product capability can change the ownership of a project.

Configuring User Capabilities

In the SAS Forecasting for Desktop, a user capability represents a specific privilege that might be granted to the user. The Administer Product capability is the most significant, because it grants the authority to manage the product and all content that is inside the product. You can use the Administer Product capability to manage product security to some degree. However, generally in the SAS Forecasting for Desktop, capabilities work more like preferences than a security measure. For example, certain design elements might be selectively hidden from a user by choice based on how the user intends to use the SAS Forecasting for Desktop clients. For more information about security, see "Understanding Security" on page 31.

In the SAS Forecasting for Desktop, you can customize the capabilities granted to a user by modifying two properties files. Each capability is associated with a unique property to which you can assign a value of true or false to indicate whether the corresponding capability is granted. Both configuration properties files use the same set of properties for convenience. The configuration files are as follows:

***!DESKTOPHOME*\userdefs.properties**

You can use this file to customize capabilities for all users. SAS Forecasting for Desktop includes a set of default capabilities that are granted to users. You can selectively override these defaults in a global manner rather than customizing them for each user individually. See "Capabilities for the SAS Forecasting for Desktop Users" on page 23 for a list of the capabilities.

***!USERHOME*\fsuser.properties**

You can use this file for all customizations that are specific to a particular user, including the ability to override granted capabilities. This file is checked last for customizations, so any setting in this file is guaranteed to be the final setting. This is the same file that is used to customize the user ID and display name to change the ownership status of a user. For more information, see "Understanding Project Ownership and User Capabilities" on page 21.

Use the following syntax to specify a capability:

capability.*key*=*value*

You must include the **capability.** prefix before the capability-specific key value in order to properly form the property name. See "Capabilities for the SAS Forecasting for Desktop Users" on page 23 for the values. Use **true** or **false** for the property value.

Capabilities for the SAS Forecasting for Desktop Users

The following tables lists the capabilities and keys for the SAS Forecasting for Desktop that can be specified in ***!DESKTOPHOME*\userdefs.properties** and ***!USERHOME*\fsuser.properties**.

Table 5.2 Client Access

Capability	Key	Description	Default Value
Forecast Studio	FSForecastStudio	Enables the use of the SAS Forecast Studio for Desktop.	**true**
Management Clients	FSForecastPlugin	Enables the use of the SAS Forecast Server Plug-in for the SAS Forecast Project Manager.	**true**

Capability	Key	Description	Default Value
Macro Bridge	FSMacroBridge	Enables the use of the SAS Forecast Batch Interface (for example, the FSCOPY macro).	`true`

Table 5.3 General Features

Capability	Key	Description	Default Value
Administer Product	FSAdministerProduct	Enables the user to administer the SAS Forecasting for Desktop. Users who are assigned this capability are restricted only by other capabilities. The ownership and sharing settings are ignored.	`false`
Analyze Time Series	FSAnalyzeTimeSeries	Enables the Series View in the SAS Forecast Studio for Desktop.	`true`
Analyze Models	FSAnalyzeModels	Enables the Modeling View in the SAS Forecast Studio for Desktop.	`true`
Change Series Usage	FSChangeSeriesUsage	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.	`true`

Capability	Key	Description	Default Value
Import New Data	FSImportNewData	Enables the user to choose whether to update the project when the SAS Forecasting for Destop determines that changes have been made to the input data set. For example, this capability selects the **Use updated data if available** check box in several dialog boxes in SAS Forecast Studio for Desktop, such as the Update Project Version dialog box and the Reforecast Project dialog box. Users who are not assigned this capability cannot update the input data set for the project. Instead, the project uses the version of the data that was available when the project was created.	`true`

Table 5.4 Forecasts

Capability	Key	Description	Default Value
Modify Forecasts	FSModifyForecasts	Enables the user to forecast the project or series in SAS Forecast Studio for Desktop.	`true`
Reconcile Forecasts	FSReconcileForecasts	Enables the user to reconcile the hierarchy. For example, this capability enables the **Reconcile Hierarchy** menu item, the Reconcile Hierarchy icon in the Forecasting View, and the **Reconcile** message (that appears in the workspace if there are override conflicts or unresolved nodes).	`true`

Capability	Key	Description	Default Value
Override Forecasts	FSOverrideForecasts	Enables the user to create, edit, or delete override values. For example, this capability enables the Override calculator, the ability to lock overrides, and ability to select **Set Scenario Forecast Values as Overrides** in the Scenario Analysis View.	`true`

Table 5.5 Models

Capability	Key	Description	Default Value
Change Model Selection	FSChangeModelSelection	Enables the 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.	`true`
Modify Models	FSModifyModels	Enables the user to create, delete, copy, and edit models. This capability enables the user to import models from a catalog and export models to a catalog.	`true`

Table 5.6 Events

Capability	Key	Description	Default Value
Change Event Usage	FSChangeEventUsage	Enables the user to specify whether an event is used in the model. For example, this capability enables the **Usage in system-generated models** menu in the Events Properties dialog box.	`true`
Modify Events	FSModifyEvents	Enables the user to create, delete, copy, and edit events. This capability enables the user to import events from a catalog and export events to a catalog.	`true`

Table 5.7 Environments

Capability	Key	Description	Default Value
View Environment Details	FSViewEnvironmentDetails	Enables the user to view the properties of SAS Forecast Server environments	`true`
Manage Environment Settings	FSManageEnvironmentSettings	Enables the user to edit the properties of SAS Forecast Server environments.	`true`
Manage Environments	FSManageEnvironments	Enables the user to create, delete, and rename environments.	`true`

Table 5.8 Projects

Capability	Key	Description	Default Value
Create Projects	FSCreateProjects	Enables the user to create a new project. This capability enables the New Project wizard.	`true`
Manage Projects	FSManageProjects	Enables the 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.	`true`
Manage Project Access	FSManageProjectAccess	Enables the user to specify whether a project is shared and to change the ownership of a project. For example, this capability enables the **New Owner** and **Share** options in the SAS Forecast Server Plug-in for SAS Management Console.	`true`
Manage Project Settings	FSManageProjectSettings	Enables the 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.	`true`

Capability	Key	Description	Default Value
View Project Scripts	FSViewProjectScripts	Enables the user to view the project code that runs 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.	`true`
Modify Project Scripts	FSModifyProjectScripts	Enables the 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 SAS Start-up and Shutdown Code dialog box.	`true`

Table 5.9 Notes

Capability	Key	Description	Default Value
View Notes	FSViewNotes	Enables the Notes panel at the bottom of the Forecasting View.	`true`
Modify Notes	FSModifyNotes	Enables the user to edit the content in the Notes panel at the bottom of the Forecasting View. This capability enables the user to create and remove notes.	`true`

Table 5.10 Scenarios

Capability	Key	Description	Default Value
View Scenarios	FSViewScenarios	Enables the Scenario Analysis View in SAS Forecast Studio for Desktop.	`true`
Modify Scenarios	FSModifyScenarios	Enables the user to create new scenarios and to edit, save, and delete existing scenarios.	`true`

Table 5.11 Special Features

Capability	Key	Description	Default Value
Legacy Features	FSEnableLegacyFeatures	Enables legacy features. This capability is provided only to help users through transitional periods that result from design changes in the next release. Support for these legacy features is limited. You should enable legacy features only if directed by SAS Technical Support.	`false`
Experimental Features	FSEnableExperimentalFeatures	Enables experimental features that are still under development. These experimental features might change or be removed in a future release. No migration support is provided for these experimental features. You should enable experimental features only if directed by SAS Technical Support.	`false`
Debugging Features	FSEnableDebuggingFeatures	Enables debugging features that are provided to help users and SAS Technical Support when problems arise. You should enable debugging features only if directed by SAS Technical Support.	`false`

Example Configuration Files

The capabilities that are applied to a particular user session are determined by taking the built-in defaults, and then applying the overrides from ***!DESKTOPHOME*\userdefs.properties** and ***!USERHOME*\fsuser.properties** (in order). Files that do not exist are simply interpreted as providing no overrides.

- Here is an example of a ***!DESKTOPHOME*\userdefs.properties** file that overrides the default capabilities assignments to also include the Administer Product capability. Because this file is common to all users, all users of this desktop configuration are treated as product administrators unless they override the setting in their **!USERHOME\fsuser.properties** file to set the value to*false*.

  ```
  capability.FSAdministerProduct=true
  ```

- Here is an example of a ***!USERHOME*\fsuser.properties** file that explicitly sets the user ID and display name. In addition, it grants the experimental features capability to the user. Because this is the last file processed for capabilities, any setting in this file is guaranteed to be the final setting.

```
user.name=sasdemo
user.display.name=Local User

capability.FSEnableExperimentalFeatures=true
```

Configuring the Desktop Mode

Desktop Mode Options

SAS Forecasting for Desktop provides a set of configurable options to control how the desktop mode operates. These options parallel those available in SAS Forecast Server deployments. In the SAS Forecasting for Desktop, the options are managed in a properties file that is located at ***!DESKTOPHOME*\config.properties**. As with capabilities, the desktop mode uses built-in default values for these settings. You can override the default values by using this optional configuration file. Unlike SAS Forecast Server deployments, all feature-related options are enabled by default, but you can disable them if desired. The following table lists the configurable options in the SAS Forecasting for Desktop and the built-in default value of each.

Table 5.12 Logging

Property	Description	Accepted Values	Default Value
forecasting.log.jdbc	Insert JDBC query details in the SAS log output. This should be enabled only at the request of technical support.	true/false	**false**
forecasting.log.filtering	Filter out internal details to simplify SAS log output. This should be disabled only at the request of technical support.	true/false	**true**

Table 5.13 Libraries

Property	Description	Accepted Values	Default Value
forecasting.enable.auto.environment.libs	Enable use of automatic (internal) environment libraries.	true/false	**true**
forecasting.enable.config.environment.libs	Enable use of configured (external) environment libraries.	true/false	**true**

Table 5.14 Scripts

Property	Description	Accepted Values	Default Value
forecasting.enable.environment.autoexec	Enable execution of user-provided SAS scripts to start up and shut down environments.	true/false	`true`
forecasting.enable.project.autoexec	Enable execution of user-provided SAS scripts to start up and shut down projects.	true/false	`true`

Table 5.15 Archiving

Property	Description	Accepted Values	Default Value
forecasting.archive.compression.level	Specify compression level for created project archives.	0 (no compression) – 9 (full compression) If no value is specified, use Java default.	no value
forecasting.archive.extensions.ignore	Ignore specified file extensions when creating project archives.	Comma-separated values list of file extensions	`sas7bndx`
forecasting.archive.extensions.cport.data	Treat specified file extensions as portable data sets when creating project archives.	Comma-separated values list of file extensions	`sas7bdat`
forecasting.archive.extensions.cport.catalog	Treat specified file extensions as portable catalogs when creating project archives.	Comma-separated values list of file extensions	`sas7bcat`
forecasting.archive.cport.options	Include specified options on the PROC CPORT statements when creating project archives.	*text*	no value

Example Configuration File

Here is an example of a ***!DESKTOPHOME*\config.properties** file that overrides the default settings to disable the execution of user-provided SAS scripts.

```
forecasting.enable.project.autoexec=false
```

Understanding Security

In the SAS Forecasting for Desktop, security elements exist in three areas:

Capabilities
: can be interpreted as security controls

Project ownership and sharing
: can limit the project owner’s ability to perform certain actions

File system permissions
: can limit access to files based on underlying operating system design. Normal users cannot adjust these permissions.

Security in the SAS Forecasting for Desktop differs from the security in a network environment. The operating systems that SAS Forecasting for Desktop runs in are designed for multiple user environments with built-in security. The three main differences between security in a desktop environment and security in a network environment are as follows:

- In the SAS Forecasting for Desktop, all security other than operating system file permissions is advisory only. Users are permitted to change their own settings. Hence, the settings are more for convenience than for control.
- The SAS Open Metadata Architecture (OMA) permissions that exist in a network environment are not present in a desktop environment.
- Capabilities assignments are controlled through local configuration files under desktop mode. More specifically, capabilities for a user are determined by applying the following three "profiles" in order:

  1. Capabilities are initialized with built-in defaults. These defaults grant all capabilities except the Administer Product and special features capabilities.
  2. Customizations are applied from `!DESKTOPHOME\userdefs.properties`. You can use these settings to customize the common defaults for all users.
  3. Customizations are applied from `!USERHOME\fsuser.properties`. Users can use these settings to customize his or her own common defaults.

For more information, see “Understanding Project Ownership and User Capabilities” on page 21.

6

Libraries in the SAS Forecasting for Desktop

Understanding Libraries

Overview of Libraries

SAS Forecasting for Desktop uses SAS libraries and data sets to manage and access project data and to manage environment libraries. Alternatively, libraries can be defined and managed within the SAS Foundation configuration files in the normal manner. The SAS Forecast Project Manager for Desktop client does not support the management of such libraries from within the client, but does report their existence.

For more information about libraries in the SAS Forecasting for Desktop, see the online Help for the SAS Forecast Project Manager for Desktop. To open this application, select **Start** ▸ **All Programs** ▸ **SAS** ▸ **SAS Forecast Project Manager for Desktop 12.1**. For more information about other SAS 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. A library name that begins with an underscore is the naming convention reserved for an internal SAS library.

External Library Assignments

Libraries that are not assigned by the SAS Forecasting for Desktop clients at run time are called external libraries.

Here are the main sources of external library assignments:

- All SAS libraries that are provided with SAS are automatically assigned to the SAS Forecasting for Desktop. Examples of Base SAS libraries are Sashelp, Sasuser, and Work.
- You can assign libraries by adding LIBNAME statements to your SAS Foundation configuration files.
- You can assign libraries by adding LIBNAME statements to the start-up code for a forecasting environment. These libraries are available only for the associated environment. If you have assigned a library in the environment's start-up code and you want to shut down the environment, you must include a corresponding LIBNAME CLEAR statement in the code.

Libraries Assigned by the SAS Forecasting for Desktop

SAS Forecasting for Desktop enables you to manage and assign libraries. Even if a library is not currently assigned, it appears in SAS Forecasting for Desktop clients like any other library. When a client detects that a library is going to be accessed (for example, when a library is referenced by a project that is being opened), the SAS Forecasting for Desktop assigns the library before allowing access.

Unlike the SAS Forecast Server, support for such libraries is available by default in the SAS Forecasting for Desktop. However, this support can be disabled if desired. For more information about these configurable options, see Table 5.13 on page 30.

Under SAS Forecasting for Desktop, client-assigned libraries must be defined within the context of a forecasting environment. Consequently, the library is available only when you are using that environment. For each environment library, you can choose to store the content in these ways:

- in a dedicated internal location. This type of library is an automatic environment library. You can use these libraries as scratch areas where working data can be stored across multiple sessions. These libraries are considered a low maintenance option because the associated content is automatically managed. For example, renaming or deleting an automatic environment library automatically affects the content in the library. For example, if you delete an automatic library, the content in that library is also deleted.

  Automatic environment libraries are implicitly defined by creating a subdirectory for each library in the `\Libraries` subdirectory in the environment file area. The name of the subdirectory implies the LIBNAME for the library assignment and the access settings for the library.

  When a client detects that an automatic environment library needs to be used, the library is assigned by using the BASE engine.

- in an external location that is specified in the library definition. This type of library is a configured environment library. These libraries enable you to manually manage your content and to control where the content is stored. For example, if you delete a configured environment library, only the library definition is deleted. You must manually delete the contents of the library.

  You can define configured environment libraries in the `\Config\libs` data set in the environment file area. Each row of this data set represents a library assignment and contains the information required for the BASE engine:

  - the LIBNAME

- □ the full path to the library directory
- □ whether access to the library should be Read-Only

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

For more information about how to create and manage these libraries in the SAS Forecasting for Desktop, see the online Help for the SAS Forecast Project Manager for Desktop. To open this application, select **Start** ▶ **All Programs** ▶ **SAS** ▶ **SAS Forecast Project Manager for Desktop 12.1**. For more information about how to manage libraries through the file system, see "How to Manually Define an Automatic Environment Library" on page 35, and "How to Manually Define a Configured Environment Library" on page 36.

Authorization Checks

A key to good library management and use is performing authorization checks on the data accesses that are attempted by users. Library access is controlled through the Read-Only setting on the assignment and the file system permissions. The Read-Only settings are stored in dictionary tables for all libraries in the SAS session. These tables are checked by the SAS Forecasting for Desktop to determine the current user's level of access to each library.

Any additional security settings that are used by libraries are not visible to the SAS Forecasting for Desktop. As a result, errors can occur because SAS Forecasting for Desktop cannot check security settings before an action is attempted. This is primarily a concern when using non-BASE library assignments within the SAS Foundation configuration files or start-up scripts.

How to Manually Define an Automatic Environment Library

Note: You can manage both environments and environment libraries using the SAS Forecast Project Manager for Desktop. For more information, see *SAS Forecast Studio: User's Guide* or the online Help for the SAS Forecast Project Manager for Desktop.

To manually define an automatic environment library :

1. Verify that the SAS Forecasting for Desktop is configured to support automatic environment libraries by checking that the value of the `forecasting.enable.auto.environment.libs` property in the *`!DESKTOPHOME`*`\config.properties` file is *true*.

2. In the `\Libraries` subdirectory in the environment file area, create a subdirectory for the automatic environment library. The name of the subdirectory implies the LIBNAME for the library assignment. For example, if the intended LIBNAME is Hpfuser and the environment file area is located at `C:\myenv`, then you would create the path `C:\myenv\Libraries\hpfuser`.

   Note: If the `Libraries` directory does not exist, then just create it.

3. Specify the security access for the library. By default, a library is assigned for Read and Write access. If the library should have Read-Only access, append

an `.r` to the directory name. For example, the `hpfuser.r` directory name creates the Hpfuser library with Read-Only access.

When a user next opens this forecasting environment in SAS Forecast Studio for Desktop, the new library is available.

How to Manually Define a Configured Environment Library

Note: You can use the SAS Forecast Project Manager for Desktop to manage both environments and environment libraries. For more information, see *SAS Forecast Studio: User's Guide* or the online Help for the SAS Forecast Project Manager for Desktop.

To manually define a configured environment library:

1. Verify that the SAS Forecasting for Desktop is configured to support configured environment libraries by checking that the value of the `forecasting.enable.config.environment.libs` property in the *`!DESKTOPHOME`*`\config.properties` file is *true*.

2. In the `\Config` subdirectory in the environment file area, create a libs.sas7bdat file. For example, if the environment file area is located at `C:\myenv`, then the file path would be `C:\myenv\Config\libs.sas7bdat`. This file (data set) should contain three character variables: LIBNAME, PATH, and READONLY.

   Note: If the libs.sas7bdat data set does not exist at run time, the SAS Forecasting for Desktop tries to create it when a user opens this forecasting environment in SAS Forecast Studio for Desktop.

3. Edit the libs.sas7bdat 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`.

When a user opens this forecasting environment in SAS Forecast Studio for Desktop, the new library is available.

7

Advanced Topics

Archiving a Project

Overview

Project archives are the primary means of managing projects. Archives can be used for a variety of purposes including the following:

- backing up projects
- migrating projects between product versions
- copying or moving projects between environments, including the transfer between the SAS Forecasting for Desktop and SAS Forecast Server deployments.

You can create and extract project archives by using either the SAS Forecast Project Manager for Desktop client or the SAS Forecast Batch Interface macros (specifically, FSIMPORT and FSEXPORT). For more information about the former, see the online Help for the SAS Forecast Project Manager. For more information about the SAS Forecast Batch Interface options, see "About the SAS Forecast Batch Interface" on page 52.

The archiving properties in the SAS Forecasting for Desktop ***!DESKTOPHOME*** **\config.properties** file enable you to control the process for creating and extracting project archives. However, customizing them is rarely appropriate. For more information about the former, see "Managing Configuration Files" on page 20.

To archive a project by using the SAS Forecast Project Manager for Desktop:

1. Open SAS Forecast Project Manager for Desktop.
2. In the navigation tree, expand the System node and select the environment that contains the project.
3. On the **Projects** tab, select the project that you want to archive and click .
4. The Archive dialog box appears.
5. Specify a name for the archive. By default, a generated name is provided (Archive_1, Archive_2, and so on). The name cannot be the name of an existing archive.
6. (Optional) Enter a description for the archive.
7. Specify the location for the archive.
8. Click **OK** .

Transferring Projects between SAS Forecasting for Desktop and SAS Forecast Server

The SAS Forecasting for Desktop suite is designed to support the exchange of projects between the desktop and SAS Forecast Server deployments. One of the simplest means of performing a project transfer is to use the SAS Forecast Batch Interface, which is included in both suites and is naturally interoperable with both types of deployment as long as they are from the same release.

You can use the SAS Forecast Batch Interface FSEXPORT macro to export the project to a project archive. The project must be exported to a shared network location in order to perform the transfer by using a single SAS script. After the project is archived, close the session to the source deployment. Then, a connection to the destination deployment is made in the same macro session, and the project is unarchived using the FSIMPORT macro.

Adding Start-Up and Shutdown Code

Understanding Start-Up or Shutdown Code

You can customize your project or environment by adding SAS code to run when it is open or closed. For example, you can enable the MPRINT system option when the project opens. As a result, any SAS statements that are generated when the user runs SAS Forecast Studio for Desktop macros are traced for debugging. When you close a project or environment, you might want SAS Forecast Studio for Desktop to remove any temporary files that were created, to back up files, or to automatically export your results to an external data set.

Note: Do not use start-up code to assign a library that contains the input data set for a project. The input data set must be available before the project is created or opened. Hence, the timing of this assignment is not appropriate for input data.

For start-up and shutdown code, the SAS session used for processing is shared whenever possible to improve performance. Consequently, shutdown code must reverse any changes that are made by the start-up code so that the session is restored to its initial state after the close action. For example, suppose you enable the MPRINT system option in the start-up code. If you then disable this option in the shutdown code, tracing is still enabled during any later projects that use that same SAS session. Such problems can lead to unusual errors that are difficult to reproduce and resolve.

Start-up code is saved in the **`startup.sas`** file, and shutdown code is saved in the **`shutdown.sas`** file. These files are in the **`\Config`** subdirectory of the associated project or environment. For convenience, the SAS Forecast Studio for Desktop and SAS Forecast Project Manager for Desktop also provide editing view for these project and environment files, respectively.

Configuring the Ability for Code to Run

Unlike SAS Forecast Server, the SAS Forecasting for Desktop enables the use of start-up or shutdown code by default. However, these options can be disabled if desired. When the option is disabled, a message appears at the top of the SAS Start-up and Shutdown Code dialog box. The message states that code execution is not permitted. The dialog boxes otherwise continue to function normally to allow the stored code to be managed. For more information about these configuration options, see Table 5.14 on page 31.

Managing Start-Up and Shutdown Code for Projects

When you create a project, you can add start-up and shutdown code using the New Project wizard in SAS Forecast Studio for Desktop . After the project is created, you can edit this code using the Project Properties dialog box in the SAS Studio for Desktop. This option is available only when the project is open.

Note: To add start-up or shutdown code in SAS Forecast Studio for Desktop, the user must be assigned the Modify Project Scripts capability.

1 Open SAS Forecast Studio for Desktop. Select **File** ▶ **Projects**. In the Projects dialog box, open the project that you want to modify.

2 Select **File** ▶ **Project Properties**. The Project Properties dialog box appears.

3 Click **Modify**. The SAS Start-up and Shutdown Code dialog box appears.

4 On the **Start-up Code** and **Shutdown Code** tabs, enter the SAS code that you want to include in the project.

   Note: If the use of this code has been disabled, a message appears at the top of the SAS Start-up and Shutdown Code dialog box. To enable the code to run, see "Configuring the Ability for Code to Run" on page 39.

5 To update the shutdown code that is saved in memory, select the **Also update the stored shutdown code to be used during the upcoming close action** check box. To view the code that is saved in memory, click **View the stored shutdown code**.

   Note: When a project is opened, the shutdown code (that runs when you close the project) is saved in memory. If you change the shutdown code, you must select the **Also update the stored shutdown code to be used during the upcoming close action** check box if you want the changes to be reflected in the saved copy of the shutdown code.

6 Click **OK** to save your changes and close the SAS Start-up and Shutdown Code dialog box. Click **OK** again to close the Project Properties dialog box.

Managing Start-Up and Shutdown Code for Environments

When you create an environment, you can add start-up or shutdown code using the SAS Start-Up and Shutdown dialog box in SAS Project Manager for Desktop.

1 Open SAS Project Manager for Desktop.

2 From the **Environments** tab, select the environment that you want to add the code to. Click 📋.

3 The SAS Start-up and Shutdown Code dialog box appears.

4 On the **Start-up Code** and **Shutdown Code** tabs, enter the SAS code that you want to include in the project.

   Note: If the use of this code has been disabled, a message appears at the top of the SAS Start-up and Shutdown Code dialog box. To enable the code to run, see "Configuring the Ability for Code to Run" on page 39.

5 Click **OK** to save your changes and close the SAS Start-up and Shutdown Code dialog box.

Creating Custom Time Intervals

Creating a Custom Time Interval

SAS Forecasting for Desktop includes numerous time intervals. However, your site might need a custom time interval.

To create a custom interval:

1 Using a DATA step program or a text 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 the raw data contains date values (the number of days since January 1, 1960), use a date value 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 a datetime value for the BEGIN variable and assign a datetime format.

     You can define the end of each period (using an END variable) and the seasonal cycle (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 the END variable for each observation is one less than the value of the BEGIN variable for 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. Calculations include forecasting and operations that might extend beyond the time series (such as filters).

   CAUTION! An error will occur if a date or datetime value (in the actual data, in a forecast, or in a plot) is outside the range of the custom interval definition. For your custom interval, the value for the BEGIN variable must start with the earliest historical date 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 value 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 the config file (`C:\Program Files\SASHome\SASFoundation\9.3\sasv9.cfg`) that is used by the SAS Forecasting for Desktop. For more information, see *SAS 9.3 System Options Reference, Second Edition*. In the SAS Forecasting for Desktop, the syntax for the INTERVALDS= system option is more restrictive than the syntax 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 to 32 characters. The name cannot contain an underscore or an embedded numeric character (with the exception that the name can end with one or more digits). For example, StoreHours12 is a valid name. Store12Hours is an invalid name. When you specify multiple custom 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 library reference 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 intervals the next time you start SAS Forecast Studio for Desktop. For example, you can select an interval when you create a new project in the New Project wizard. If a custom interval that you created is not available in the list, check the SAS log for errors in the definition of the custom interval.

Example 1: Creating a StoreHours Custom Interval

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

1 Log on to the computer that the SAS Forecasting for Desktop is installed on. Use a text editor to add the following code to the **C:\Program Files\SASHome\SASFoundation\9.3\sasv9.cfg** file:

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

2 Start a SAS session. In this SAS session, complete these steps:

   a Assign the LIBNAME CustIntLib to the directory in which the data set (in this example, StoreHoursDS) for the custom interval 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 dates of the first and last observations in the data set.

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

- Use the WEEKDAY function to determine the day of the week for a specific date. By default, the day of the week is 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 datetime value.

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

Example 2: Creating a Custom Interval Definition Interactively

This example shows how to create a custom interval interactively.

1 Start a SAS session on the computer that the SAS Forecasting for Desktop is installed on.

2 Click the New Library icon in the toolbar. The New Library dialog box appears.

3 For **Name**, enter the LIBNAME that you specified in the INTERVALDS= system option.

4 For **Path**, click **Browse**, and select the path where the custom interval will be stored. Click **OK** to close the Select dialog box. Click **OK** again to close the New Library dialog box.

5 In the SAS Explorer, right-click the library that you 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 top of column A, and select **Column Attributes**.

8 Specify these properties:

- Change **Name** from `A` to **`BEGIN`**.
- Select **Numeric** for **Type**.
- Enter **`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 in cells as needed. When you are finished, close **VIEWTABLE**. Click **Yes** to save your changes. In the Save As dialog box, select the library that you entered in step 3.

11 For **Member Name**, enter the name of the data set that you specified in the INTERVALDS= system option.

12 Click **Save** to save the data set.

Creating a Custom Format

SAS Forecasting for Desktop includes numerous formats. However, you might have formats at your site that are not available in SAS Forecast Studio for Desktop. Using the FORMAT procedure, you can create custom formats and make them available. For more information about this procedure, see the *Base SAS Procedures Guide*.

To create a custom format:

1 In PROC FORMAT, specify where to store the custom format using the LIBRARY= option. To store the format in the `C:\myfmts` location, 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—is 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 that is used by the SAS Forecasting for Desktop.

a Open the configuration file in the following default location:

Windows
`C:\Program Files\SASHome\SASFoundation\9.3\sasv9.cfg`

b In the sasv9.cfg file, use the SET system option to define the library. In the following code, the SET system option associates the FSFMTS catalog with the **\myfmts** library. Set the FMTSEARCH system option to FSFMTS so that the SAS Forecasting for Desktop searches this library.

```
-set FSFMTS "C:\myfmts"
-insert fmtsearch FSFMTS
```

The SAS Forecasting for Desktop resolves any references to custom formats that are stored in **\myfmts**.

Creating a Customized List of Events

Creating a Customized Lists of Events

SAS Forecast Studio for Desktop provides a list of predefined events. However, you might need to add to this list or create a customized list. To create an event, you must use the HPFEVENTS procedure first, and then use the DATEKEYS procedure to associate a date key, label, and locale with this event. The procedure results are saved in a data set. The date key forms the name of the event. The EVENTDS= system option is defined (preferably in the SAS configuration file) to point to these event data sets so that the events appear in SAS Forecast Studio for Desktop.

Note: The DATEKEYS procedure is a part of SAS Time Series Studio 12.1 that is experimental for this release.

To create a customized list:

1. Using a DATA step program or a text editor, create the data set that defines the events.

2. To define the name and location of your event data set, specify the EVENTDS= system option in the SAS Foundation configuration file (**C:\Program Files\SASHome\SASFoundation\9.3\sasv9.cfg** . For more information, see *SAS 9.3 System Options Reference, Second Edition*.

The customized list of events should appear in the list of events the next time you start SAS Forecast Studio for Desktop.

Example: Creating Super Bowl Events

1. Start a SAS session , and run the following code:

```
proc datekeys;
     datekeydef SuperBowl =             '15JAN1967'D  '14JAN1968'D  '12JAN1969'D  '11JAN1970'D
                          '17JAN1971'D  '16JAN1972'D  '14JAN1973'D  '13JAN1974'D  '12JAN1975'D
                          '18JAN1976'D  '09JAN1977'D  '15JAN1978'D  '21JAN1979'D  '20JAN1980'D
                          '25JAN1981'D  '24JAN1982'D  '30JAN1983'D  '22JAN1984'D  '20JAN1985'D
                          '26JAN1986'D  '25JAN1987'D  '31JAN1988'D  '22JAN1989'D  '28JAN1990'D
                          '27JAN1991'D  '26JAN1992'D  '31JAN1993'D  '30JAN1994'D  '29JAN1995'D
                          '28JAN1996'D  '26JAN1997'D  '25JAN1998'D  '31JAN1999'D  '30JAN2000'D
                          '28JAN2001'D  '03FEB2002'D  '26JAN2003'D  '01FEB2004'D  '06FEB2005'D
                          '05FEB2006'D  '04FEB2007'D  '03FEB2008'D  '01FEB2009'D  '07FEB2010'D
                          '06FEB2011'D  '05FEB2012'D  '03FEB2013'D  '02FEB2014'D
                         / PULSE=DAY locale= 'en_US'
```

```
                           label="Super Bowl Sunday";
     datekeykey Christmas    / locale= 'en_US'
                               label="Christmas Day";
     datekeykey Thanksgiving / locale= 'en_US'
                               label="Thanksgiving Day in US";
     datekeykey TurkeyDay=N4W5NOVYR / locale= 'en_US'
                               label="Alternate Name for Thanksgiving";
     datekeydata out=holiday condense;
run;
proc print data=holiday;
run;



options eventds=(holiday);

title 'LIST option gives a list of available datekeys- with defaults';
proc datekeys;
     datekeydata out=holidaylist LIST;
run;
proc print data=holidaylist;
run;

title 'LIST option gives a list of available datekeys - without defaults';
proc datekeys;
     datekeydata out=holidaylist LIST NODEFAULTS;
run;
proc print data=holidaylist;
run;


data ts(keep=date);
     do i=1 to 120;
        date=INTNX('MONTH','01JAN2000'D,i-1);
        output;
     end;
     format date DATE.;
run;
title 'Using system option custom datekeys are allowed';
proc hpfevents;
     id date interval=month;
     eventkey SuperBowl;
     eventkey Xmas=Christmas;
     eventdata out=myevents condense;
     eventdummy out=mydummies;
run;
proc print data=myevents;
run;
proc print data=mydummies;
run;
```

2 Open the SAS configuration file in the following default location:

Windows

`C:\Program Files\SASHome\SASFoundation\9.3\sasv9.cfg`

3 In the sasv9.cfg file, use the EVENTSDS= system option to point to the events data set.

```
EVENTSDS=("c:\eventslist")
```

Starting SAS Forecast Studio for Desktop with Options

Open a Specific Forecasting Environment or Project in SAS Forecast Studio for Desktop

To open a specific forecasting environment or project in SAS Forecast Studio for Desktop, include **`launchFile=`*`filename`*** in the launching command line, where ***`filename`*** is the path to an .fs file that contains the parameters of a specific forecasting environment or project. For example, suppose you create the file **`C:\Europe\Sales\Products2012.fs`**. The **`Products2012.fs`** file contains the following code:

```
environment=Default
project=Products2012
```

When you run SAS Forecast Studio for Desktop using `launchFile=C:\Europe\Sales\Products2012.fs`, SAS Forecast Studio for Desktop attempts to open the Default environment. If the launch is successful, SAS Forecast Studio for Desktop attempts to open the Products2012 project. If the specified environment or project does not exist, the client will simply ignore the bad option and start up normally.

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 for Desktop uses to detect the time interval. By default, SAS Forecast Studio for Desktop uses the first 10,000 observations. Usually, a few distinct time ID values are needed to detect the time interval. However, a data set with many repeated values of the time ID variable sometimes requires a larger sample of observations to get enough distinct values. Increasing the number of observations might improve time-interval detection when you are using data sets that have many repeated values. However, values of 100,000 observations or larger could cause noticeably slower performance when you are creating a new project.

To configure SAS Forecast Studio for Desktop to use a specified number of observations:

1 Open the forecaststdw.ini file in a text editor. For example, in a default Windows installation, this file is located in the **`C:\Program Files\SASHome\SASForecastStudioforDesktop\12.1`** directory.

2 In the .forecaststdw.ini file, enter a new `JavaArgs_`*`n`* entry, where *`n`* is the number of the next argument in the list. Increment the argument number accordingly (for example, **`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 for Desktop, you can specify a default value for the sample size. The sample is used to check the validity of BY variable values and formats. The default value is applied 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 the sample size, the first *n* observations of the data set are used to answer the following questions:

- 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 of these questions is “yes,” then an appropriate error dialog box is displayed. The project is not created or saved.

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

1. Open the forecaststdw.ini file in a text editor. For example, in a default Windows installation, this file is located in the **`C:\Program Files\SASHome\SASForecastStudioforDesktop\12.1`** directory.
2. In the forecaststdw.ini file, enter a new `JavaArgs_`*`n`* entry, where *`n`* is the number of the next argument in the list. Increment the argument number accordingly (for example, **`JavaArgs_14=-Dcom.sas.analytics.forecasting.by_validation_sample=75000`**).

Part 4

SAS Forecast Batch Interface

8

Using the SAS Forecast Batch Interface

About the SAS Forecast Batch Interface

Overview

The SAS Forecast Batch Interface is available for the SAS Forecast Server and SAS Forecasting for Desktop. Macros in the SAS Forecast Batch Interface provide a SAS language interface to the SAS Forecast Server and SAS Forecasting for Desktop. Effectively, the macros serve as an alternative client to SAS Forecast Studio, SAS Forecast Studio for Desktop, the SAS Forecast Server Plug-ins for SAS Management Console, SAS Forecast Project Manager, and SAS Forecast Project Manager for Desktop. Some macros enable you to create projects with options identical to those available in SAS Forecast Studio and SAS Forecast Studio for Desktop. However, most of the macros focus on the management of environments and projects.

Installation and Availability

When your site licenses the SAS Forecast Server or SAS Forecasting for Desktop, the SAS Forecast Batch Interface is automatically included in all SAS Foundation installations. The SAS Forecast Batch Interface component is installed in the extension directory **`forecastbat`**. On Windows, the macros in the SAS Forecast Batch Interface reside in an autocall library in **`!SASROOT/forecastbat/sasmacro`**. On UNIX, they reside in the **`SASFoundation/9.3/sasautos`** directory.

General Usage

Whether you are working with the SAS Forecast Server or SAS Forecasting for Desktop, the basic pattern of usage is the same. Here are the general usage steps:

1. Call the FSLOGIN macro to begin a product session.
2. Call one or more of the other SAS Forecast Batch Interface macros to perform actions in the product session.
3. Call the FSLOGOUT macro to end the product session.

For example, the FSGETENV macro can be used with the PRINT=YES argument to retrieve the list of available forecasting environments and print them to the SAS LISTING output. To make this call, the code would be similar to the following:

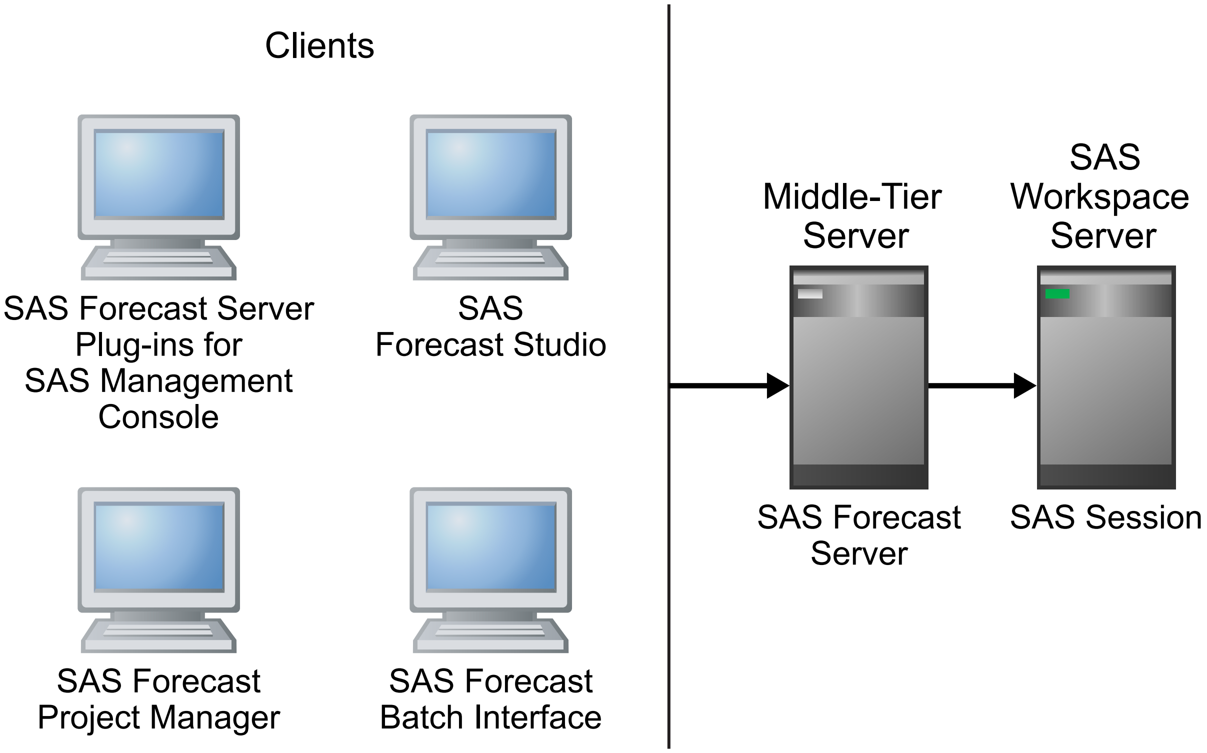
```
%FSLOGIN(arguments);
%FSGETENV(print=YES);
%FSLOGOUT();
```

In this example, the arguments for FSLOGIN are intentionally omitted because they depend on the type of deployment. There is more information about these arguments in the subsections that follow.

Note: Prior to release 12.1, each SAS Forecast Batch Interface macro created and ended a dedicated product session during the course of its call. This required FSLOGIN information to be included in the arguments for each macro. As of release 12.1, these macro arguments are no longer supported. SAS scripts that were created for releases prior to 12.1 must be updated to use FSLOGIN and FSLOGOUT macros instead.

Working with SAS Forecast Server

The following figure shows how the SAS Forecast Batch Interface macros interact with the other components of SAS Forecast Server:


Clients
SAS Forecast Server Plug-ins for SAS Management Console
SAS Forecast Studio
SAS Forecast Project Manager
SAS Forecast Batch Interface
Middle-Tier Server
SAS Workspace Server
SAS Forecast Server
SAS Session

As with the other SAS Forecast Server clients, the SAS Forecast Batch Interface uses SAS environments to identify the SAS deployments that are available. As a result, the same configuration requirements apply to the SAS Forecast Batch Interface. SAS Forecast Batch Interface macros must have access to a sas-environment.xml file that defines the list of SAS environments or an error message is displayed. For the login macro, FSLOGIN, the SASENVIRONMENT argument is used to identify the selected SAS environment by name. As a result, 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. The symbolic name is used by the clients that do not have a graphical user interface because a symbolic name is independent of your locale. (Do not use the DESC= attribute to specify the SAS environment. The DESC= attribute is the label that appears when the SAS environment is displayed in a list, but is a locale-sensitive value.)

Note: FSLOGIN is the only macro that refers to SAS environments. All other environments in the SAS Forecast Batch Interface are forecasting environments.

SAS Forecast Server deployments also require credential-based authentication when creating a session. The arguments “user” and “password” are used to provide authentication information. It is not recommended that you use plain-text passwords in your SAS code. Instead, use PROC PWENCODE to create an encrypted password.

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

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

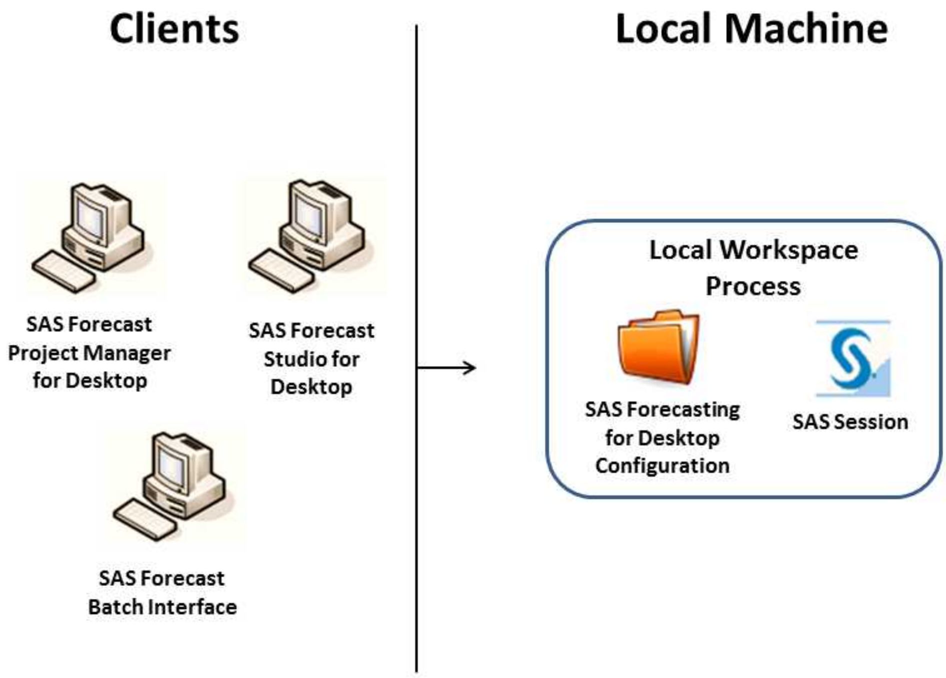
Copy the encrypted password from the SAS log, and paste the password into your macro code. For more information about the PWENCODE procedure, see the *Base SAS 9.3 Procedures Guide*.

For example, suppose your site has only one SAS environment named DEFAULT and the SAS Forecast Server is included. Running the previous example with the SASDEMO identity and an encrypted password {sas002}123 require the following code:

```
%FSLOGIN(sasEnvironment=default, user=sasdemo, password={sas002}123);
%FSGETENV(print=YES);
%FSLOGOUT();
```

Work with SAS Forecasting for Desktop

The following figure shows how the SAS Forecast Batch Interface macros interact with the other components of SAS Forecasting for Desktop.


Clients
Local Machine
SAS Forecast Project Manager for Desktop
SAS Forecast Studio for Desktop
SAS Forecast Batch Interface
Local Workspace Process
SAS Forecasting for Desktop Configuration
SAS Session

Because SAS Forecasting for Desktop can use your operating system identity directly, creating a product session is trivial. The FSLOGIN call needs to indicate that only a desktop session is requested, which is done by passing the DESKTOP=YES argument. Here is the complete code for the previous example for SAS Forecasting for Desktop:

```
%FSLOGIN(desktop=YES);
%FSGETENV(print=YES);
%FSLOGOUT();
```

Summary of Macros

Note: The SAS Forecasting for Desktop license does not permit the use of the FSCREATE macro. Instead, projects must be created using the SAS Forecast Studio for Desktop client. In addition, the FSRUNRPT macro does not apply to SAS Forecasting for Desktop deployments because stored processes are not available in desktop mode.

The following table lists the macros in the SAS Forecast Batch Interface:

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 project.
FSCREATE	Creates a new project in batch mode. **Note:** If you license SAS Forecast Studio for Desktop, the FSCREATE macro is not available. It is available only if you license the SAS Forecast Server. For more information about SAS Forecast Studio for Desktop, see the *SAS Forecast Studio: User's Guide*.
FSDELARC	Deletes an archived project.
FSDELENV	Deletes an existing forecasting environment.
FSDELEVT	Deletes specified events from the event repository.
FSDELPRJ	Deletes an existing project.
FSEVTREQ	Sets the required attributes for events in the project’s event repository.
FSEXPALL	Archives all projects.
FSEXPORT	Archives a single project.
FSEXPSET	Exports the project settings to a local file.
FSGETENV	Shows all of the details about a forecasting environment.

Macro Name	Description
FSGETPRJ	Shows all of the details about a project.
FSGETURP	Lists the names of unregistered projects in a specified forecasting environment.
FSIMPALL	Imports a collection of projects from the project archives.
FSIMPORT	Imports a single project from the project archives.
FSLOAD	Opens an existing SAS Forecast Server project and loads global macro variables that describe the project.
FSLOGIN	Creates a new session for a specified instance of the middle tier.
FSLOGOUT	Closes a session of a specified instance of the middle tier.
FSLOOP	Iterates through a data set and invokes a callback macro with the values from each row.
FSMIGALL	Migrates all existing SAS Forecast Server projects to the current version of the SAS Forecast Server.
FSMIGPRJ	Migrates an existing SAS Forecast Server project to the current version of the SAS Forecast Server.
FSMOVE	Moves a SAS Forecast Server project to a different forecasting environment.
FSNEWENV	Creates a new SAS Forecast Server forecasting environment.
FSREGENV	Registers an existing directory structure as a forecasting environment. You can also use this macro to register projects in the environment.
FSREGPRJ	Registers a project.
FSREN	Renames a project.
FSRUNPRJ	Runs an existing project at a specified stage.
FSRUNRPT	Runs a forecasting report and saves the results to a local directory. **Note:** FSRUNRPT only works with SAS Forecast Server deployments.
FSSETDAT	Changes the input data source for a project.
FSSETDEF	Stores a default value for a macro argument.

Macro Name	Description
FSSETOWN	Assigns the owner of a project.
FSSETPUB	Changes the public access (sharing) status of a project.
FSUNREG	Unregisters an existing project.
FSUNRENV	Unregisters a forecasting environment.
FSUPDATE	Updates the settings of an existing SAS Forecast Studio project.
FSVER	Returns the version of the SAS Forecast Batch Interface.

The FSCREATE and FSRUNRPT macros are not applicable to SAS Forecasting for Desktop.

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(EVENTNAME=,EVENTTYPE= [,options]);

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSADDEVT macro if no default values were previously stored. The required arguments are separated by commas.

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

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 arguments and are separated by commas.

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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 separated by spaces.

EVENTCOMBINATIONRULE = ADD | MAX | MIN | MINNZ | MINMAG | MULT

specifies which combination rule to use when the event has several values that overlap in the same time period. Here are 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 | 1 | 0

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 date and 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 | 1 | 0

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 | 1 | 0
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 value 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.

EVENTPERIODICITYOCCURRENCES = *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 the SAS Forecast Server automatically fits a model. The default value is NO. Here are the valid values:

NO specifies that the event should 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 event should be included in the model as long as the event parameters are significant.

YES specifies that the event should be included in the model as long as the model can be diagnosed.

UNDEF specifies that the event should 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 are separated by spaces. Here are the valid values:

- Dates in the ddmmmyyyy format (for example, 01JAN2012).
- Datetime 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*.

PROJECTNAME = *project-name*

specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

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

&FSADDEVT = SUCCESS|ERROR

Example

```
%fsaddevt(projectname=Project1,
          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();

Example

```
%fsclear();
```

FSCOPY Macro

The FSCOPY macro copies a project.

Syntax

%FSCOPY (SOURCEENVIRONMENT=, DESTINATIONENVIRONMENT=, SOURCEPROJECT=, DESTINATIONPROJECT=[options]) ;

Details

Required Arguments

You must specify either a source environment and a destination environment or a source project and a destination project. You cannot specify both an environment and a project at the same time. Required arguments are separated by commas.

SOURCEENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server environment to be used as the source. This option is required only if there are multiple environments available to the user. If this option is omitted, the current stored default value is used. The default value is Default. A global default value for SOURCEENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

DESTINATIONENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server environment to be used as the destination. This option is required only if there are multiple environments available to the user. If this option is omitted, the current stored default value is used. The default value is Default. A global default value for DESTINATIONENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

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. If this option is omitted, then the current stored default value is used. A global default value for the SOURCEPROJECTNAME can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

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. If this option is omitted, then the current stored default value is used. A global default value for the DESTINATIONPROJECTNAME can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

Options

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

ARCHIVEFOLDER = *directory-name*

specifies the directory in which to save the archive. It is recommended that you specify a value for this directory. The ARCHIVEFOLDER= and REMOTEARCHIVEFOLDER= options are closely related.

- If you specify the ARCHIVEFOLDER= option, but you do not specify a value for the REMOTEARCHIVEFOLDER= option, SAS Forecast Server assumes that the directory for the archive folder and the remote archive folder is the same. For this to work, the source and destination environments must be using the same server. If these environments are on different servers, an error results.
- If you specify a value for the REMOTEARCHIVEFOLDER= option, you must specify a value for the ARCHIVEFOLDER= option.
- If you do not specify a value for the ARCHIVEFOLDER= option or the REMOTEARCHIVEFOLDER= option, both options are set to the Work directory in the source environment. This option is valid only when you are copying or moving a project from one destination to another on the same server. If the source and destination environments are on different servers, an error results.

CPORT = TRUE | FALSE | YES | NO | 1 | 0

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 NO.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

REMOTEARCHIVEFOLDER = *directory-name*

specifies the directory in which the archived projects of the source SAS Workspace Server can be found on the destination SAS Workspace Server (for example, `\\sourceserver\SAS\ForecastStudio\Archives`). If you specify a remote archive folder, you must specify a value for the ARCHIVEFOLDER= option. For more information about how the REMOTEARCHIVEFOLDER= and ARCHIVEFOLDER= options work together, see the description for the ARCHIVEFOLDER= option.

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

TMPARCHIVENAME = *project-name*

specifies the name of the temporary archived project. The default is _fs_tmp_archive. The name must not match an existing archived project.

Results

The FSCOPY global macro variable indicates whether the FSCOPY macro terminated successfully or encountered errors:

&FSCOPY= SUCCESS | ERROR

Example

```
%fscopy(sourceprojectname=prdat1,
destinationprojectname=prdat2,
sourceEnvironment=Default,
destinationEnvironment=Default,
);
```

FSCREATE Macro

The FSCREATE macro creates a new project in batch mode. If you license SAS Forecast Studio for Desktop, the FSCREATE macro is not available. It is available only if you license the SAS Forecast Server.

Syntax

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

Details

If you license SAS Forecast Studio for Desktop, the FSCREATE macro is not available. It is available only if you license the SAS Forecast Server. For more information about SAS Forecast Studio for Desktop, see the *SAS Forecast Studio: User's Guide*.

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSCREATE macro if no default values were previously stored. 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. If you do not specify this option, the current stored default value for PROJECTNAME is used. The value of this default cannot be specified with the FSSETDEF macro.

DATA = *SAS-data-set-name*
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 variables can be specified only if HIERARCHY= NO | FALSE. Multiple variable names are separated by spaces.

Options

The following options can be used with the FSCREATE macro. Options must follow 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 of all variables, specify **`ACCUMULATE=AVG`**.
- To use a total accumulation of all variables except for VAR1, which will use an average accumulation, specify **`ACCUMULATE=TOTAL AVG(var1).`**
- To use a total accumulation of all variables except VAR1 and VAR2, which will use an average accumulation, and VAR3, which will accumulate the observations based on standard deviation, specify **`ACCUMULATE=TOTAL AVG(var1 var2) STD(var3).`**

ADJUST = *var1*(*var2var3*) [/operation=(pre,post)]
specifies the adjustment options for the dependent variables. If this option is omitted, the current stored 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 | 1 | 0
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 | 1 | 0
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 are separated by a space.

COMBINE = YES | TRUE | NO | FALSE | 1 | 0

specifies whether the 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 model. 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 valid values for the selection criterion:

- AADJRSQ—Amemiya's adjusted R-square
- ADJRSQ—adjusted R-square
- 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 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-square
- RWRSQ—random walk R-square
- 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 ARIMAX option, the UCM option, or both the ARIMAX 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. 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 ARIMAX option, the UCM option, or both the ARIMAX 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 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 residuals 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, equally restricted least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast residuals subject to the constraint 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 constraint 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 time series across the combination, one or more combination contributors can have a missing value. This option determines the treatment of missing values in the final combination for time series 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 1.

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 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 HORMISSPCT 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 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 ARIMAX option, the UCM option, or both the ARIMAX 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 | 1 | 0

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 ARIMAX option, the UCM option, or both the ARIMAX and UCM options are set to YES or TRUE.

ENDZEROSMAXNUM = *integer*
specifies the maximum number of trailing zeros for a nonzero model. The default value is 0.

ENDZEROSMAXPCT = *integer*
specifies the maximum percentage of trailing zeros for a nonzero model relative to the number of nonzero values in the entire series. The default value is 0.

ENDZEROSMINOBS = *integer*
specifies a threshold for the series length that is required to enable the ENDZERO test. The default value is 0.

ENVIRONMENT = *environment-name*
specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

ESM = YES | TRUE | NO | FALSE | 1 | 0
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 value must be the name of a BY variable. By default, no level is specified for fitting exponential smoothing models.

HIERARCHY = YES | TRUE | NO | FALSE | 1 | 0
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 are from 1 to 100.

This option is different from the COMBINEMISSPCT option, which is for the estimation region. The COMBINEMISSPCT option and the HORMISSPCT 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.

HOLDOUTPCT = *n* | NO | FALSE
specifies the maximum percentage of the series length to be used as the holdout sample for model selection. The holdout percentage guards against errors that are caused by holdout samples that are too large for the selected time series. For example, suppose a series has a length of 100 periods (excluding any leading or trailing missing values). If the HOLDOUT= option is set to 20, and the HOLDOUTPCT= option is set to 10, the holdout sample contains 10 periods.

If you do not specify a value for the HOLDOUTPCT= option, the default value is 5.

IDFORMAT = *date*

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

IMPORTSETTINGSPATH = *full-path-to-file*

specifies the file that contains the project settings to use in the new project. You must specify the full path to the macro file on the machine that invokes the macro. For example, this could be a file created by %FSEXPSET. When you set the IMPORTSETTINGSPATH= option, the project settings in the file are used to initialize the values for the project. If you set any project settings by using arguments in the FSCREATE macro, these arguments are processed after the IMPORTSETTINGSPATH= option. This enables project settings to be adjusted individually after the initial import of the project settings from the file.

INPUT = *variable*

specifies the input (or independent) variables. Variable names are separated by spaces.

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 are the 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 a date interval to create a 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 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 ARIMA models. The default value is 2.

MAXPCTOUTLIERS = *n*

specifies the maximum number of outliers to include in ARIMA models specified 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 the SAS Forecast Server uses the smaller value. For example, suppose you set MAXNUMOUTLIERS=5 and MAXPCTOUTLIERS=10. The maximum number of the outliers is 5 or 10% of the series length, whichever is smaller.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

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-name*

specifies a model selection list that contains models that you can use. The default value is SASHELP.HPFDFLT.TSFSSELECT.

PUBLICACCESS = YES | TRUE | NO | FALSE | 1 | 0

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 the SAS Forecast Server reconciles the hierarchy. By default, the 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(REGIONAME).
- 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. Variable names are separated by spaces.

RUN = YES | TRUE | NO | FALSE | 1 | 0

specifies whether to produce forecasts when the project is created. The default value is YES. The NO value is equivalent to "Save SAS code to produce forecasts later" in SAS Forecast Studio. This means that the SAS code (.sas file in the project folder) for the task (for example, CREATE_PROJECT_IMPORT_DATA.sas) should be submitted when forecasts are needed. If the FSRUN macro is submitted without first submitting this code, it will automatically select the task and submit the code.

SASENVIRONMENT = *environment-name*

specifies the symbolic name of the SAS environment where the middle tier for the 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, the 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 specify 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 seasonality 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—Each missing value is set to 0.
- MISSING—Each missing value is set to missing. This is the default value.
- AVG—Each missing value is set to the accumulated average value.
- MIN—Each missing value is set to the accumulated minimum value.
- MED—Each missing value is set to the accumulated median value.
- MAX—Each missing value is set to the accumulated maximum value.
- FIRST—Each missing value is set to the accumulated first nonmissing value.
- LAST—Each missing value is set to the accumulated last nonmissing value.
- PREV—Each missing value is set to the previous accumulated nonmissing value. Missing values at the beginning of the accumulated series remain missing.
- NEXT—Each missing value is set to the next accumulated nonmissing value. 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 in the INTERVAL= option. You can specify the multiplier in the INTERVAL option. For example, if you specify 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 can specify the offset in the INTERVAL option. For example, if you specify INTERVAL=WEEK.2, then the time interval is WEEK, and the offset 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 are the valid values:

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 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. If the project is hierarchical with reconciliation, only NONE is allowed.

Here are the valid values:

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 | 1 | 0

specifies whether the 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. This option is valid only 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 are the valid values:

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 terminated successfully or encountered 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 can be used across multiple projects. For example, you can use SAS 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,
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 project.

Syntax

%FSDELARC(ARCHIVENAME= [,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSDELARC macro if no default value was previously stored.

ARCHIVENAME = *archived-project-name*

specifies the filename of the archive that you want to delete the project from (for example, myarchive.far). If the archived project name does not end with .far, then the extension is automatically assumed.

Options

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

ARCHIVEFOLDER = *archive-folder-name*
: specifies the directory containing the archived project to delete. If this option is not specified, the default location is assumed.

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project that is associated with the archived project. You must specify this option only if you do not specify the ARCHIVEFOLDER= option. The value of this default cannot be specified with the FSSETDEF macro.

Results

The FSDELARC global macro variable indicates whether the FSDELARC macro terminated successfully or encountered errors:

&FSDELARC = SUCCESS | ERROR

Example

```
%fsdelarc(projectname=pd1,
archivename=ArchPD1.far,
);
```

FSDELENV Macro

The FSDELENV macro deletes an existing forecasting environment. When you delete an environment, the registration and file system content are deleted.

Syntax

%FSDELENV(ENVIRONMENT= [,options]) ;

Details

Required Arguments

The following argument is required with the FSDELENV macro.

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. This option has no default value. The value of this default cannot be specified with the FSSETDEF macro.

Options

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

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

The FSDELENV global macro variable indicates whether the FSDELENV macro terminated successfully or encountered errors:

&FSDELENV = SUCCESS | ERROR

Example

```
%fsdelenv(enviroment=Default
);
```

FSDELEVT Macro

The FSDELEVT macro deletes specified events from the event repository.

Syntax

%FSDELEVT(EVENTNAMES= [,options]);

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSDELEVT macro if no default value was previously stored..

EVENTNAMES = *name1 name2*
specifies the events that you want to delete. Only one event name is required. Event names are separated by spaces. Event names are case sensitive.

Options

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

ENVIRONMENT = *environment-name*
specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

EVENTINTEGRITYRULE = NONE | CHECK | FORCE
determines how events are deleted when those events are in combination events 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.

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSDELEVT global macro variable indicates whether the FSDELEVT macro terminated successfully or encountered errors:

&FSDELEVT= SUCCESS | ERROR

Example

```
%fsdelevt(projectname=Project1,
         eventnames=evt2 evt4,
         environment=Default
         );
```

FSDELPRJ Macro

The FSDELPRJ macro deletes an existing project. You can also delete any related archived projects in the default archive location.

Syntax

%FSDELPRJ(PROJECTNAME= [,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSDELPRJ macro if no default value was previously stored.

PROJECTNAME = *project-name*

specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this argument, then the default value for PROJECTNAME is used. The value of this default cannot be specified with the FSSETDEF macro.

Options

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

DELETEARCHIVES = YES | TRUE | NO | FALSE | 1 | 0

specifies whether the archives in the default archive location are to be deleted. By default, all archives are deleted with the project.

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

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 terminated successfully or encountered errors:

&FSDELPRJ = SUCCESS | ERROR

Example

```
%fsdelprj(projectname=prdat4,
deleteArchives=YES,
);
```

FSEVTREQ Macro

The FSEVTREQ macro sets the required attributes for events in the project’s event repository.

Syntax

%FSEVTREQ(EVENTNAME=, EVENTREQUIRED=[,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSEVTREQ macro if no default values were previously stored. The required arguments are separated by commas.

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 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.

Options

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

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSEVTREQ global macro variable indicates whether the FSEVTREQ macro terminated successfully or encountered errors:

&FSEVTREQ = SUCCESS | ERROR

Example

```
%fsevtreq(projectname=prdata1,
eventname=_ALL_,
eventrequired=undef
);
```

FSEXPALL Macro

The FSEXPALL macro archives all projects. This macro should be used only by users who are assigned the Administer Product capability in SAS Management Console.

Syntax

%FSEXPALL([options]) ;

Details

Options

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

ARCHIVEFOLDER = *directory-name*
: specifies the directory in which to save the archives. If this option is not specified, the archives are saved to project subfolders in the default location.

CPORT = YES | TRUE | NO | FALSE | 1 | 0
: exports data sets and catalogs using CPORT. You must specify this option if you unarchive the project on a different platform. The default is NO.

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

OUT = *SAS-data-set-name*
: specifies the fully qualified name of a SAS data set that contains information about the archived projects. The format is *library.dataset.*

SUBFOLDERS = YES | TRUE | NO | FALSE | 1 | 0
: specifies whether archives are saved in subfolders corresponding to project names. If this option is set to NO, all archives are saved in the same location. To use this option, you must specify the ARCHIVEFOLDER= option. The default is NO.

PRINT = YES | TRUE | 1 | NO | FALSE | 0
: specifies whether to print the output data set. The default is NO.

Results

The FSEXPALL global macro variable indicates whether the FSEXPALL macro terminated successfully or encountered 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 was created.
MODIFIED	specifies the date and time when the project was last modified.
CREATEDBY	specifies the user ID that created the project.

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 while archiving the project.

In addition, archives for all projects are created in the default archive location. The name of the archived project is the name of the project followed by the NAMESUFFIX= parameter.

Example

```
%fsexpall(out=work.projects,
namesuffix=_fs_export31,
environment=default
);
```

FSEXPORT Macro

The FSEXPORT macro archives a single project.

Syntax

%FSEXPORT(ARCHIVENAME=, DESCRIPTION=, [options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSEXPORT macro if no default values were previously stored. The required arguments are separated by commas.

ARCHIVENAME = *archived-project-name*
specifies the filename of the archive that you want to export the project into (for example, myarchive). Do not specify the file extension. The file extension is automatically assumed to be .far. Use the ARCHIVEFOLDER= option to specify the directory of the archive file.

DESCRIPTION = *text*
specifies a description for the archive. The description is recorded in the Manifest.fs file in the archive. If a comma is in the description string, the description must be enclosed in quotation marks.

Options

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

ARCHIVEFOLDER = *directory-name*
: specifies the directory in which to save the archive. If this option is not specified, the archive is saved in the default location.

CPORT = YES | TRUE | NO | FALSE | 1 | 0
: exports data sets and catalogs using CPORT. You must specify this option if you unarchive the project on a different platform. The default is NO.

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

Results

The FSEXPORT global macro variable indicates whether the FSEXPORT macro terminated successfully or encountered errors:

&FSEXPORT = SUCCESS | ERROR

An archived project with the ARCHIVENAME= filename is created in the default location.

Example

```
%fsexport( projectname=pd1,
archivename=ArchPD1.far,
description=Project pd1
);
```

FSEXPSET Macro

The FSEXPSET macro exports the project settings to a local file.

Syntax

%FSEXPSET(OUTPATH= [,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSEXPSET macro if no default values were previously stored.

OUTPATH = *full-path-of-output-file*
specifies the path and filename of the local file. This file has an .fps extension.

Options

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

ENVIRONMENT = *environment-name*
specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

Results

The FSEXPSET global macro variable indicates whether the FSEXPSET macro terminated successfully or encountered errors:

&FSEXPSET = SUCCESS | ERROR

Example

```
%fsexpset(outpath=C:\ForecastServer\Projects\Settings.fps,
environment=Default,
projectname=Project3,
);
```

FSGETENV Macro

The FSGETENV macro shows all of the details about a forecasting environment.

Syntax

%FSGETENV([options]) ;

Details

Options

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

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

OUT = *SAS-data-set-name*
specifies the fully qualified name of a SAS data set that contains the program results. The default value is Work.FSEnv.

PRINT = YES | TRUE | 1 | NO | FALSE | 0
specifies whether to print the output data set. The default is NO.

Results

The FSGETENV global macro variable indicates whether the FSGETENV macro terminated successfully or encountered 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 SAS 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 the SAS Forecast Server that the environment uses.

Example

```
%fsgetenv(out=work.getprojects,
print=YES
);
```

FSGETPRJ Macro

The FSGETPRJ macro shows all of the details about a project.

Syntax

%FSGETPRJ([options]) ;

Details

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

OUT = *SAS-data-set-name*
: specifies the fully qualified name of a SAS data set that contains the program results. The default value is Work.FSPrj.

PRINT = YES | TRUE | 1 | NO | FALSE | 0
: specifies whether to print the output data set. The default is NO.

Results

The FSGETPRJ global macro variable indicates whether the FSGETPRJ macro terminated successfully or encountered errors:

&FSGETPRJ=SUCCESS | ERROR

The OUT= option produces a SAS data set that contains the following variables:

NAME	specifies the project name.
DESCRIPTION	specifies the description of the project.
VERSION	specifies the version of the SAS Forecast Server for the project.
LIBRARY	specifies the library for the project's input data set.

DATASET specifies the name of the project’s input data set.

CREATED specifies the date and time when the project was created.

MODIFIED specifies the date and time when the project was last modified.

CREATEDBY specifies the project identity of the person who created the project initially which in some cases is the same as the user ID.

MODIFIEDBY specifies the project identity of the last person who modified the project which in some cases is the same as the user ID.

OWNEDBY specifies the project identity of the project's owner which in some cases is the same as the user ID.

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,
print=YES
);
```

FSGETURP Macro

The FSGETURP macro lists the names of any unregistered projects in a specified forecasting environment. To register the projects, you can use the project names generated by the FSGETURP macro as input to the FSREGPRJ macro.

Syntax

%FSGETURP([options]) ;

Details

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and

should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

OUT = *SAS-data-set-name*
specifies the fully qualified name of a SAS data set that contains the program results. The default value is Work.FSPrj.

PRINT = YES | TRUE | 1 | NO | FALSE | 0
specifies whether to print the output data set. The default is NO.

Results

The FSGETURP global macro variable indicates whether the FSGETURP macro terminated successfully or encountered errors:

&FSGETURP = SUCCESS | ERROR

Example

```
%fsgeturp(environment=Default,
out=sales.unregistered,
print=YES
);
```

FSIMPALL Macro

The FSIMPALL macro imports a collection of projects from the project archives. This macro should be used only by a user who has the Administer Product capability.

Syntax

%FSIMPALL(PROJECTDS=, ARCHIVEFOLDER=[options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSIMPALL macro if no default values were previously stored. The required arguments are separated by commas.

PROJECTDS = *SAS-data-set-name*
specifies the fully qualified name of a SAS data set that contains the list of the SAS Forecast Server projects to import. The format is *library.dataset*. The PROJECTDS= data set is created when you specify the OUT= option in the FSEXPALL macro.

The PROJECTDS= data set must be a SAS data set that contains the following variables:

NAME specifies the project name.

ARCHIVENAME specifies the name of the archive.

ARCHIVEFAIL equals 1 if an error was detected during archiving. If the ARCHIVEFAIL variable is included, only the archives where the value of the ARCHIVEFAIL variable is 0 are imported.

ARCHIVEFOLDER = *directory-name*

specifies the directory that contains the archived projects (for example, `C:\SAS\ForecastStudio\Archives).`

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see “FSSETDEF Macro” on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

SUBFOLDERS = YES | TRUE | NO | FALSE | 1 | 0

specifies whether archives are stored in subfolders corresponding to project names in the archive folder location. If this option is set to NO, all archives are assumed to be stored in the default archive folder location. The default value is NO.

Results

The FSIMPALL global macro variable indicates whether the FSIMPALL macro terminated successfully or encountered 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 more information, see “FSMIGALL Macro” on page 97.

Example

```
%fsimpall(projectds=work.projects,
archivefolder=\\sourceserver\SAS\ForecastStudio\Archives
);
```

FSIMPORT Macro

The FSIMPORT macro imports a single project from the project archives.

Syntax

%FSIMPORT(ARCHIVEPATH= [,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSIMPORT macro if no default value was previously stored.

ARCHIVEPATH = *directory-name*
specifies the location of the archived project.

Options

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

ENVIRONMENT = *environment-name*
specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSIMPORT global macro variable indicates whether the FSIMPORT macro terminated successfully or encountered errors:

&FSIMPORT = SUCCESS | ERROR

Example

```
%fsimport(projectname=prdat2,
archivepath=C:\SAS\ForecastStudio\Archives\pd1\ArchPD1.far
);
```

FSLOAD Macro

The FSLOAD macro opens an existing SAS Forecast Server project and loads global macro variables that describe the project.

Syntax

%FSLOAD([options]) ;

Details

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSLOAD global macro variable indicates whether the FSLOAD macro terminated successfully or encountered 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 can be used across multiple projects. For example, you can use SAS code to explore and use data in your SAS Forecast Server projects.

Example

```
%fsload(projectname=pd1,
environment=default
);
```

FSLOGIN Macro

The FSLOGIN macro creates a new session for a specific instance of the middle tier. Each middle tier is permitted to have, at most, one session at any given time. This allows the middle tier to be used to identify the session.

Note: FSLOGIN stores information about the most recent session in a global variable for use by other macros. You can specify the middle tier explicitly by using the MIDTIER= argument.

Syntax

%FSLOGIN(DESKTOP=,USER=, PASSWORD=[,options]) ;

Details

Required Arguments

The following arguments are required with the FSLOGIN macro if no default values were previously stored. The required arguments are separated by commas.

DESKTOP = YES | NO
specifies whether the specific instance of the middle tier is being run in the SAS Forecast Server or in SAS Forecasting for Desktop. This argument is used only the first time a session is created for that instance of the middle tier. By default, this argument is set to NO, and the new session is created for the SAS Forecast Server.

USER = *user-name*
specifies the user name that you use to log on to SAS Forecast Studio. This argument is required when DESKTOP=NO, which is the default setting.

PASSWORD = *password*
specifies the password that you use to log on to SAS Forecast Studio. This argument is required when DESKTOP=NO, which is the default setting.

Options

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

SASENVIRONMENT = *environment-name*
specifies the symbolic name of the SAS environment. This symbolic name is specified by the NAME= attribute in the sas-environment.xml file and is case sensitive. The default value is Default.

MIDTIER = *label*
identifies which middle tier to use. When you specify this option, you must give each middle tier a unique label (name) so that a later macro call can be directed to the correct session. A name should never be reused for a different middle tier during the same SAS session, even after previous SAS sessions are closed.

This option is typically omitted and the default value is used. For the SAS Forecast Server, the default label is fsmain.

Results

The FSLOGIN global macro variable indicates whether the FSLOGIN macro terminated successfully or encountered errors:

&FSLOGIN=SUCCESS | ERROR

Example

```
%fslogin(user=sasdemo,
password=Password1,
sasEnvironment=default,
desktop=NO
);
```

FSLOGOUT Macro

The FSLOGOUT macro closes a session of a specific instance of the middle tier.

Syntax

%FSLOGOUT([options]);

Details

Options

The following option can be used with the FSLOGOUT macro. Options must follow arguments.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

The FSLOGOUT global macro variable indicates whether the FSLOGOUT macro terminated successfully or encountered errors:

&FSLOGOUT=SUCCESS | ERROR

FSLOOP Macro

The FSLOOP macro iterates through a data set and invokes a callback macro with the values from each row.

Syntax

%FSLOOP(DATA=,CALLBACK=,VARLIST=[,options]);

Details

Required Arguments

The following arguments are required with the FSLOOP macro if no default values were previously stored. The required arguments are separated by commas.

DATA = *data-set-name*
specifies the data set that contains the arguments that you want to use.

CALLBACK = *name-of-handler-macro*
specifies the handler macro to invoke for each row of arguments. The macro must implement the expected signature.

VARLIST = *var-name1 var-name2 . . . var-name_n*
specifies the variables in the data set to pass to the calls to the handler macro. Variable names must be separated by spaces. The handler macro is expected to accept each row variable through a macro variable of the same name, unless the VARARGS= optional argument is used to map the variable names to other names of macro arguments.

Options

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

ARGDATA = *data-set-name*
specifies the data set that contains additional fixed argument values. These arguments are passed to each generated macro call with the arguments from the input data set. The ARGDATA= data set must be formatted for two character columns named Name and Value. By default, no fixed arguments are used.

VARARGS = *macro_varname1 macro_varname2...macro_varname_n*
specifies the variable names for the macro input that corresponds to the variable names in the VARLIST= argument. If you do not specify the VARARGS= argument, the argument names for the handler macro are assumed to match the variable names in the input data set. For example, if VARLIST=NAME DESCRIPTION, but the macro expects the arguments to be named X= and Y=, then you should specify VARARGS= X Y for the FSLOOP macro.

Results

The FSLOOP global macro variable indicates whether the FSLOOP macro terminated successfully or encountered errors:

&FSLOOP=SUCCESS | ERROR

Example

```
%FSLOGIN(sasenvironment=default,
user=sasdemo,
password=Password1
);

* Example of using FSGETPRJ to get project information
and put results in work.prjlst;
```

```
%FSGETPRJ(
  environment=Default,
  out=work.prjlst);

* Example using arguments data set to fill extra parameters;
data args;
  length name $32;
  length value $128;

  name='environment';
  value='Default';
  output;

  name='isPublic';
  value='true';
  output;
run;

%FSLOOP(
  data=work.prjlst,
  callback=FSSETPUB,
  varlist=name,
  varargs=project,
  argdata=args
);

%FSLOGOUT();
```

FSMIGALL Macro

The FSMIGALL macro migrates all existing SAS Forecast Server projects to the current version of the SAS Forecast Server.

Syntax

%FSMIGALL(OUT= [,options]) ;

Details

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS

Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

OUT = *SAS-data-set-name*
specifies the fully qualified name of a SAS data set that contains information about the archived projects. The format is *library.dataset*. The default value is Work.FSMig.

PRINT = YES | TRUE | 1 | NO | FALSE | 0
specifies whether to print the output data set. The default is NO.

Results

The FSMIGALL global macro variable indicates whether the FSMIGALL macro terminated successfully or encountered 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 was created.
MODIFIED	specifies the date and time when the project was 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(out=work.migrated,
print=YES
);
```

FSMIGPRJ Macro

The FSMIGPRJ macro migrates an existing SAS Forecast Server project to the current version of the SAS Forecast Server.

Syntax

%FSMIGPRJ([options]) ;

Details

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSMIGPRJ global macro variable indicates whether the FSMIGPRJ macro terminated successfully or encountered errors:

&FSMIGPRJ=SUCCESS | ERROR

Example

```
%fsmigprj(projectname=pd1,
environment=default
);
```

FSMOVE Macro

The FSMOVE macro moves a project to a new destination.

Syntax

%FSMOVE([options]) ;

Details

Options

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

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

CPORT = YES | TRUE | NO | FALSE | 1 | 0
: exports data sets and catalogs using CPORT. This option is needed if the source and destination are running on different operating systems. The default is NO.

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. If this option is omitted, the current stored default value is used. The default value is Default. A global default value for DESTINATIONENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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. A global default value for DESTINATIONPROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

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. If this option is omitted, the current stored default value is used. The default value is Default. A global default value for SOURCEENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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. A global default value for SOURCEPROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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

Results

The FSMOVE global macro variable indicates whether the FSMOVE macro terminated successfully or encountered errors:

&FSMOVE=SUCCESS | ERROR

Example

```
%fsmove(sourceprojectname=prdat1,
destinationprojectname=prdat2,
remoteArchiveFolder=C:\SAS\ForecastStudio\Archives,
sourceEnvironment=Default,
destinationEnvironment=Default,
);
```

FSNEWENV Macro

The FSNEWENV macro creates a new SAS Forecast Server forecasting environment.

Syntax

%FSNEWENV(ENVIRONMENT=,PATH= [,options]) ;

Details

Required Arguments

The following arguments are required with the FSNEWENV macro if no default values were previously stored. The required arguments are separated by commas.

ENVIRONMENT = *environment-name*

specifies the name of the forecasting environment. The value for this argument is case sensitive. The default environment is Default. The value of this default cannot be customized with the FSSETDEF macro.

PATH = *file path*

specifies the absolute file path to the base content directory.

Options

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

DESC = *text*

specifies the environment description.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

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 folder. This option is used only when you are creating an environment for a networked middle tier.

WSSERVER = *workspace-server-name*
specifies the name of the logical SAS Workspace Server. The default value is SASAPP - Logical Workspace Server. This argument is ignored by SAS Forecasting for Desktop because it is assumed that the local machine is the host.

Results

The FSNEWENV global macro variable indicates whether the FSNEWENV macro terminated successfully or encountered errors:

&FSNEWENV=SUCCESS | ERROR

Example

```
%fsnewenv(environment=Default,
desc=A new environment,
path=C:\FSEnvironment,
reportspath=/MyContent/Reports,
wsserver=SASApp - Logical Workspace Server
);
```

FSREGENV Macro

The FSREGENV macro registers an existing directory structure as a forecasting environment. You can also use this macro to register all projects in the environment.

Syntax

%FSREGENV(ENVIRONMENT=, PATH=[,options]) ;

Details

Required Arguments

The following arguments are required with the FSREGENV macro if no default values were previously stored. The required arguments are separated by commas.

ENVIRONMENT = *environment-name*
specifies the name of the forecasting environment. The value for this argument is case sensitive. The default environment is Default. The value of this default cannot be customized with the FSSETDEF macro.

PATH = *file-path*
specifies the absolute file path to the base content directory on the specified SAS Workspace Server.

Options

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

DESC = *text*
specifies the environment description.

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

REGISTERPROJECTS = YES | TRUE | NO | FALSE | 1 | 0
specifies whether to register in the metadata any existing projects in the environment. The default value is YES.

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.

WSSERVER = *workspace-server-name*
specifies the name of the logical SAS Workspace Server. The default value is SASAPP - Logical Workspace Server. This argument is ignored by SAS Forecasting for Desktop because it is assumed that the local machine is the host.

Results

The FSREGENV global macro variable indicates whether the FSREGENV macro terminated successfully or encountered errors:

&FSREGENV=SUCCESS | ERROR

Example

```
%fsregenv(environment=Default,
desc=A new environment,
path=C:\FSEnvironment,
reportspath=/MyContent/Reports,
wsserver=SASApp - Logical Workspace Server,
registerprojects=no
);
```

FSREGPRJ Macro

The FSREGPRJ macro registers a project.

Syntax

%FSREGPRJ(PROJECTNAME= [options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSREGPRJ macro if no default values were previously stored. 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. If you do not specify this argument, then the current stored default value for PROJECTNAME is used. The value of this default cannot be customized with the FSSETDEF macro.

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

The FSREGPRJ global macro variable indicates whether the FSREGPRJ macro terminated successfully or encountered errors:

&FSREGPRJ=SUCCESS | ERROR

Example

```
%fsregprj(projectname=ProjectA,
environment=Default
);
```

FSREN Macro

The FSREN macro renames a project.

Syntax

%FSREN(NEWPROJECTNAME=[options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSREN macro if no default value was previously stored.

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

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

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 variable indicates whether the FSREN macro terminated successfully or encountered errors:

&FSREN=SUCCESS | ERROR

Example

```
%fsren(projectname=prdat3,
newprojectname=prdat4
);
```

FSRUNPRJ Macro

The FSRUNPRJ macro runs an existing project at a specified stage.

Syntax

%FSRUNPRJ(METHOD=,IMPORTDATA=[,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSRUNPRJ macro if no default values were previously stored. The required arguments are separated by commas.

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 | 1 | 0

specifies whether to update the data in the project with any new data in the input data set. The default value is NO. Any changes to the input data set are not applied to the project.

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

LEAD = *n*

specifies the number of periods into the future in which multi-step forecasts are made. The default value is 12.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*

specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

RETAINCHOOSE = YES | TRUE | NO | FALSE | 1 | 0

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 variable indicates whether the FSRUNPRJ macro terminated successfully or encountered 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 can be used across multiple projects. For example, you can use SAS code to explore and use data in your SAS Forecast Server projects.

Example

```
%fsrunprj(projectname=pd1,
method=FORECAST,
importdata=NO,
retainchoose=YES
);
```

FSRUNRPT Macro

The FSRUNRPT macro runs a forecasting report and saves the results to a local directory. FSRUNRPT only works with SAS Forecast Server deployments;

Syntax

%FSRUNRPT(REPORTPATH=,OUTPATH=[,options]);

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following arguments are required with the FSRUNRPT macro if no default values were previously stored. The required arguments are separated by commas.

OUTPATH = *directory-path*
specifies the local directory in which to save the output. An example is **C:\temp**.

REPORTPATH = *directory-path*
specifies the path in SAS Folders for the report definition. An example is **/MyContent/Reports/MyReport**.

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

ODSDEST = *ODS-destination*
: specifies the ODS format for the output. You can specify any valid ODS destination (for example, HTML, RTF, PDF, and so on). The default format is HTML.

ODSSTYLE = *ODS-style*
: specifies the ODS style to apply to the output. The default style is Journal.

OUTFILE = *name-of-output-file*
: specifies the name of the file that contains the output. By default, the name is assigned by the SAS Forecast Server.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

PROMPTDATA = *data-set-name*
: specifies the data set that contains the parameter values for prompts. The data set must be formatted for two character columns named Name and Value. The Name column should contain the prompt name to identify the prompt (rather than the display label that is used for the prompt). To determine the prompt name, see the report definition in SAS Management Console. By default, no prompt values are provided, so if you run a report that requires prompt values, the report will fail.

PROMPTVALUEDELIM = *delimiter*
: specifies the delimiter used when providing values for multi-value prompts. The default delimiter is a space.

SAVELOG = YES | TRUE | NO | FALSE | 1 | 0
: specifies whether to save the log from the report. The log is saved in the same location as the output files. The default value is FALSE.

UNIT = *directory-path*
: specifies the path to the focus unit in the project. By default, the root unit has the focus.

UNITDELIM = *delimiters*
: specifies the delimiters for the unit path. By default, the delimiters are ~ and |.

Results

The FSRUNRPT global macro variable indicates whether the FSRUNRPT macro terminated successfully or encountered errors:

&FSRUNRPT=SUCCESS |ERROR

Example

```
%fsrunrpt(reportpath=Samples/Getting Started/Introduction
/Introduction to Stored Processes,
outpath=C:\temp,
environment=Default,
projectname=pdsale4,
odsdest=RTF
);
```

FSSETDAT Macro

The FSSETDAT macro changes the input data source for a project.

Syntax

%FSSETDAT(DATA= [,options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSSETDAT macro if no default value was previously stored.

DATA = *input-data-set*
: specifies the new input data set. The value must be in the *libname*.*memname* format.

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and

should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSSETDAT global macro variable indicates whether the FSSETDAT macro terminated successfully or encountered errors:

&FSSETDAT= SUCCESS | ERROR

Example

```
%fssetdat(data=fslib.fsprj,
projectname=Project3,
midtier=server1,
environment=Default
);
```

FSSETDEF Macro

The FSSETDEF macro stores a default value for a macro argument.

Syntax

%FSSETDEF(MIDTIER=,KEY=,VALUE=);

Details

Required Arguments

The following arguments are required with the FSSETDEF macro if no default values were previously stored. These required arguments are separated by commas.

KEY = *key-name*
specifies the key to associate with the value specified in the VALUE variable. This key specifies what type of variable for which the default is being set. Here are the valid values:

ENVIRONMENT	specifies that the value is for an environment variable (for example, ENVIRONMENT, SOURCEENVIRONMENT, and DESTINATIONENVIRONMENT).
PROJECT	specifies that the value is for a project variable (for example, PROJECTNAME, SOURCEPROJECTNAME, and DESTINATIONPROJECTNAME).

MIDTIER = *label*
identifies which middle tier can use this stored default. The special * value can be used to indicate a default for all middle tiers. A * default is used only when the middle tier has not been assigned an explicit default value. If you are not explicitly setting the MIDTIER= value, the value automatically assigned by FSLOGIN should be passed. For more information, see "MIDTIER = label" on page 94.

VALUE = *default-argument-value*
specifies the value to store with the key. This value is used as the default argument value as appropriate by macros that support that configurable default.

Options

There are no options for the FSSETDEF macro.

Results

The FSSETDEF global macro variable indicates whether the FSSETDEF macro terminated successfully or encountered errors:

&FSSETDEF=SUCCESS | ERROR | WARNING

Example

```
%fssetdef(midtier=fsmain,
key=projectname,
value=sales
);
```

FSSETOWN Macro

The FSSETOWN macro assigns the owner of a project.

Syntax

%FSSETOWN (OWNER= [options]) ;

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSSETOWN macro if no default value was previously stored.

OWNER = *user-name*
specifies the new owner of the project.

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*

specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSSETOWN global macro variable indicates whether the FSSETOWN macro terminated successfully or encountered errors:

&FSSETOWN=SUCCESS| ERROR

Example

```
%fssetown(projectname=Project3,
owner=sastrust,
environment=Default
);
```

FSSETPUB Macro

The FSSETPUB macro changes the public access (sharing) status of a project.

Syntax

%FSSETPUB(ISPUBLIC= [options]);

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSSETPUB macro if no default value was previously stored.

ISPUBLIC = TRUE | FALSE | YES | NO | 1 | 0
: specifies whether the project should be available to all users.

Options

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

ENVIRONMENT = *environment-name*
: specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*
: identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

PROJECTNAME = *project-name*
: specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

Results

The FSSETPUB global macro variable indicates whether the FSSETPUB macro terminated successfully or encountered errors:

&FSSETPUB=SUCCESS| ERROR

Example

```
%fssetpub(project=Project3,
ispublic=YES,
environment=Default
);
```

FSUNREG Macro

The FSUNREG macro unregisters an existing project. The project files are not deleted. You can register the project again by using the project files and the FSREGPRJ macro.

Syntax

%FSUNREG(PROJECTNAME=[,options]);

Details

Required Arguments

Note: Some required arguments do not have to be specified by the user in the macro call for the requirement to be satisfied. In some cases, when a user does not specify a required argument, the macro attempts to resolve it with a default value previously stored in the macro run-time environment. If the required argument has a default value (either the system default value or a value set with FSSETDEF), then the argument does not have to be specified in the macro.

The following argument is required with the FSUNREG macro if no default value was previously stored.

PROJECTNAME = *project-name*

specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this argument, then the current stored default value for PROJECTNAME is used. The value of this default cannot be customized with the FSSETDEF macro.

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

MIDTIER=*label*

identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

The FSUNREG global macro variable indicates whether the FSUNREG macro terminated successfully or encountered errors:

&FSUNREG=SUCCESS | ERROR

Example

```
%fsunreg(projectname=prdat2,
environment=Default
);
```

FSUNRENV Macro

The FSUNRENV macro unregisters a forecasting environment. Unregistering an environment removes the registration for the environment and all projects in that environment without deleting the content in the file system. You can use the FSREGENV macro variable to re-register an environment and its projects.

Syntax

%FSUNRENV(ENVIRONMENT=[,options]) ;

Details

Required Arguments

The following argument is required with the FSUNRENV macro if no default value was previously stored.

ENVIRONMENT = *environment-name*
specifies the name of the forecasting environment. This option has no default value. The value of this default cannot be specified with the FSSETDEF macro.

Options

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

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

Results

The FSUNRENV global macro variable indicates whether the FSUNRENV macro terminated successfully or encountered errors:

&FSUNRENV=SUCCESS | ERROR

Example

```
%fsunrenv(environment=Default,
midtier=server1
);
```

FSUPDATE Macro

The FSUPDATE macro updates the settings of an existing SAS Forecast Studio project.

Note: Most of the arguments in the FSCREATE macro are also in the FSUPDATE macro. The exceptions are BY, DATA, HIERARCHY, ID, RUN, and VAR. DATA can be changed using the FSSETDAT macro.

Syntax

%FSUPDATE([options]);

Details

Options

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

ACCUMULATE = TYPE [TYPE(*var1var2)* TYPE(*var3var4*) ...]

specifies the accumulation options for the dependent, input, and reporting variables.

> **TIP** In SAS Forecast Server 12.1, **Set accumulation to the value used for aggregation** is checked by default. If the ACCUMULATE= or AGGREGATE= arguments are used, **Set accumulation to the value used for aggregation** is not selected. This setting persists, so the next time the project is opened in SAS Forecast Server 12.1, **Set accumulation to the value used for aggregation** will still not be selected.

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 of all variables, specify **`ACCUMULATE=AVG`**.
- To use a total accumulation of all variables except for VAR1, which will use an average accumulation, specify **`ACCUMULATE=TOTAL AVG(var1).`**
- To use a total accumulation of all variables except VAR1 and VAR2, which will use an average accumulation, and VAR3, which will accumulate the observations based on standard deviation, specify **`ACCUMULATE=TOTAL AVG(var1 var2) STD(var3).`**

ADJUST = *var1*(*var2var3*) [/operation=(pre,post)]
specifies the adjustment options for the dependent variables. If this option is omitted, the current stored 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.

> **TIP** In SAS Forecast Server 12.1, **Set accumulation to the value used for aggregation** is checked by default. If the ACCUMULATE= or AGGREGATE= arguments are used, **Set accumulation to the value used for aggregation** is not selected. This setting persists, so the next time the project is opened in SAS Forecast Server 12.1, **Set accumulation to the value used for aggregation** will still not be selected.

ALLOWNEGATIVE = YES | TRUE | NO | FALSE | 1 | 0
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 | 1 | 0
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.

COMBINE = YES | TRUE | NO | FALSE | 1 | 0
specifies whether the 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 valid values for the selection criterion:

- AADJRSQ—Amemiya’s adjusted R-square
- ADJRSQ—adjusted R-square
- 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 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-square
- RWRSQ —random walk R-square
- 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 ARIMAX option, the UCM option, or both the ARIMAX 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. 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 ARIMAX option, the UCM option, or both the ARIMAX 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 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 residuals 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, equally restricted least squares to compute the combination weights. The results minimize the l_2 norm of the combined forecast residuals subject to the constraint 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 constraint 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 time series across the combination, one or more combination contributors can have a missing value. This option determines the treatment of missing values in the final combination for time series 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 1.

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 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 HORMISSPCT 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 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 ARIMAX option, the UCM option, or both the ARIMAX 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 | 1 | 0
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 ARIMAX option, the UCM option, or both the ARIMAX and UCM options are set to YES or TRUE.

ENDZEROSMAXNUM = *integer*

specifies the maximum number of trailing zeros for a nonzero model. The default value is 0.

ENDZEROSMAXPCT = *integer*

specifies the maximum percentage of trailing zeros for a nonzero model relative to the number of nonzero values in the entire series. The default value is 0.

ENDZEROSMINOBS = *integer*

specifies a threshold for the series length that is required to enable the ENDZERO test. The default value is 0.

ENVIRONMENT = *environment-name*

specifies the name of the SAS Forecast Server forecasting environment. The value for this option is case sensitive. The default environment is Default. A global default value for ENVIRONMENT can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

ESM = YES | TRUE | NO | FALSE | 1 | 0

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 value must be the name of a BY variable. By default, no level is specified for fitting exponential smoothing ESM models.

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 are 1 to 100.

This option is different from the COMBINEMISSPCT option, which is for the estimation region. The COMBINEMISSPCT option and the HORMISSPCT 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.

HOLDOUTPCT = *n* | NO | FALSE

specifies the maximum percentage of the series length to be used as the holdout sample for model selection. The holdout percentage guards against errors that are caused by holdout samples that are too large for the selected time series. For example, suppose a series has a length of 100 periods (excluding any leading or trailing missing values). If the HOLDOUT= option is set to 20, and the HOLDOUTPCT= option is set to 10, the holdout sample contains 10 periods.

If you do not specify a value for the HOLDOUTPCT= option, the default value is 5.

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

INPUT = *variable*
specifies the input (or independent) variables. Variable names are separated by spaces.

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 are the 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 a date interval to create a 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 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 ARIMA models. The default value is 2.

MAXPCTOUTLIERS = *n*
specifies the maximum number of outliers to include in ARIMA models specified 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 the SAS Forecast Server uses the smaller value. For example, suppose you set MAXNUMOUTLIERS=5 and MAXPCTOUTLIERS=10. The maximum number of the outliers is 5 or 10% of the series length, whichever is smaller.

MIDTIER=*label*
identifies which middle tier session to use. This option is needed only when you are using multiple sessions in the same script which is usually unnecessary and

should be avoided where possible. If the MIDTIER= option is not specified, SAS Forecast Server uses the most recently created session if it is still open. If no such session exists or the session has been closed, then an error is reported.

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 = *data-set-name*
specifies a model selection list that contains models that you can use. The default value is SASHELP.HPFDFLT.TSFSSELECT.

PROJECTNAME = *project-name*
specifies the name of the SAS Forecast Server project. The name must be a valid SAS name. If you do not specify this option, the current stored default value for the project name is used. A global default value for PROJECTNAME can be specified with the FSSETDEF macro. For more information, see "FSSETDEF Macro" on page 110.

PUBLICACCESS = YES | TRUE | NO | FALSE | 1 | 0
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 the SAS Forecast Server reconciles the hierarchy. By default, the 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. Variable names are separated by spaces.

SEASONALITY = *n*
specifies the length of a season. When possible, the 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 specify 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 seasonality 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 value is set to 0.
- MISSING—Each missing value is set to missing. This is the default value.
- AVG—Each missing value is set to the accumulated average value.
- MIN—Each missing value is set to the accumulated minimum value.
- MED—Each missing value is set to the accumulated median value.
- MAX—Each missing value is set to the accumulated maximum value.
- FIRST—Each missing value is set to the accumulated first nonmissing value.
- LAST—Each missing value is set to the accumulated last nonmissing value.
- PREV—Each missing value is set to the previous accumulated nonmissing value. Missing values at the beginning of the accumulated series remain missing.
- NEXT—Each missing value is 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 in the INTERVAL= option. You can specify the multiplier in the INTERVAL option. For example, if you specify 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 can specify the offset in the INTERVAL option. For example, if you specify INTERVAL=WEEK.2, then the time interval is WEEK, and the offset 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 are the valid values:

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 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. If the project is hierarchical with reconciliation, only NONE is allowed.

Here are the valid values:

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 | 1 | 0

specifies whether the 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. This option is valid only 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 are the valid values:

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 FSUPDATE global macro variable indicates whether the FSUPDATE macro terminated successfully or encountered errors:

&FSUPDATE = 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 can be used across multiple projects. For example, you can use SAS code to explore and use data in your SAS Forecast Server projects.

Example

```
%fsupdate(projectname=pd1,
reconciliation=YES,
publicaccess=NO
);
```

FSVER Macro

The FSVER macro returns the version of the SAS Forecast Batch Interface.

Syntax

%FSVER();

Details

Required Arguments

There are no arguments for the FSVER macro.

Results

The output from this macro is FSVERSION, which lists the version number.

Part 5

Appendixes

Appendix 1

Using the SAS Forecasting for Desktop Utility Macros

About the SAS Forecasting for Desktop Server Utility Macros

Note: SAS Forecasting for Desktop utility macros are available only if you are licensed for SAS High-Performance Forecasting.

Utility macros can be used to manage your SAS Forecast Server projects.

Dictionary

HPF_ImportEventsFromExcel

The HPF_ImportEventsFromExcel macro imports an Excel file that contains event descriptions into an event definition data set.

Note: SAS/ACCESS Interface to PC Files must be licensed to run this macro.

Syntax

%HPF_ImportEventsFromExcel(INFILE=,SHEET=,DBMS=,
DATASET= [,options])

Details

Required Arguments

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

INFILE=*Excel-filename*

specifies the name of the Excel file that contains the event descriptions. Only one worksheet from an Excel file can be imported at a time. If the Excel file contains multiple worksheets, then multiple macro calls are required to import all of the worksheets.

SHEET=*worksheet-name*

specifies the name of the worksheet in the Excel file that contains the event descriptions. This option is not required if the Excel file specified in the INFILE option contains only one worksheet.

DBMS=*identifier*

specifies the type of data to import. This option is not required if the file extension of the Excel file is XLS or XLSX.

- If the DBMS= option is not set and the file extension of the Excel file is XLS, then DBMS=XLS is implied.
- If the DBMS= option is not set and the file extension of the Excel file is XLSX, then DBMS=XLSX is implied.

For more information about the DBMS= option, see “The IMPORT Procedure” in the *Base SAS Procedures Guide*.

DATASET=*SAS-data-set-name*

specifies the event repository (which is saved as a SAS data set) that will contain the event descriptions for use in the EVENTDATA IN= statement of PROC HPFEVENTS or in the INEVENT= option of PROC HPFDIAGNOSE or PROC HPFENGINE.

Options

You can use the following options with the HPF_ImportEventsFromExcel macro. Options must follow arguments and are separated by commas.

DATE=*column-name*

specifies the name of the column in the Excel file that contains the date information. Date information is used for the _STARTDATE_ variable of the event definition data set.

- If a value is not specified for DATE=, then the _STARTDATE_ column is used if it exists.
- If a _STARTDATE_ column does not exist, then the STARTDATE column is used if it exists.
- If neither a _STARTDATE_ nor a STARTDATE column exists, then the DATE column is used if it exists.
- _STARTDATE_ is not a required variable in an event definition data set. The date information can be specified in the _STARTDT_, _STARTOBS_, or the _KEYNAME_ column. However, each event should have a valid value for the _STARTDATE_, _STARTDT_, _STARTOBS_, or _KEYNAME_ variable for the resulting event definition data set to be processed by PROC HPFEVENTS.

ORIGINAL=OUTLOOK

specifies that the Excel file was created by exporting data from a Microsoft Outlook calendar. In this case, the date is formatted as text. It requires special input formatting.

DESCRIPTION=*column-name*

specifies the name of the column to use to create an event name.

- If DESCRIPTION= is not specified, then the value of the _NAME_, NAME, or DESCRIPTION column is used. Columns are listed in order of precedence.
- If the _LABEL_ column does not exist, then the value in the DESCRIPTION= column is used for the _LABEL_ column.
- For the _NAME_ variable, the DESCRIPTION= column is processed to form a valid SAS variable name for event processing. For the _LABEL_ variable, the _LABEL_ column is used as specified. For example, `New Years Day` creates an event named `NewYearsDay` with the label `New Years Day` because spaces are not allowed in SAS variable names (but they are allowed in labels).
- If no column value exists for the DESCRIPTION= value, then the event definition data set cannot be created.

Other Columns Processed

If any of the following columns exist in the Excel file, then they are processed as SAS variables for the event definition data set: _CLASS_, _KEYNAME_, _ENDDATE_, _DATEINTRVL_, _STARTDT_, _ENDDT_, _DTINTRVL_, _STARTOBS_, _ENDOBS_, _OBSINTRVL_, _TYPE_, _VALUE_, _PULSE_, _DUR_BEFORE_, _DUR_AFTER_, _SLOPE_BEFORE_, _SLOPE_AFTER_, _SHIFT_, _TCPARM_, _RULE_, _PERIOD_, _LABEL_. The values of the columns are used as observations in the corresponding SAS variables for PROC HPFEVENTS. This makes it possible to export an event definition data set into Excel, and then import the Excel file back into SAS.

Results

The HPF_RETURN_CODE global macro indicates whether the HPF_ImportEventsFromExcel macro finished successfully or encountered errors.

&HPF_RETURN_CODE=SUCCESS|ERROR

Example

```
%HPF_ImportEventsFromExcel(infile=C:\importEvents\Chinese2010.xlsx,
          dataset=c
          );
```

Appendix 2

Troubleshooting SAS Forecasting for Desktop

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 A2.1 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 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
- the SAS Forecasting for Desktop release number

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 capabilities 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 clients 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

To view the log in SAS Forecast Studio for Desktop, select **Tools** ▸ **SAS Log**.

To include additional information in the log, you can use the forecasting.log.jdbc and forecasting.log.filtering properties in the ***!DESKTOPHOME***`\config.properties` For more information, see "Configuring the Desktop Mode" on page 30.

Troubleshooting SAS Forecasting for Desktop

Project Owner Cannot Access Project

If you cannot access a project that you should own, you might want to check the owner information that is stored with the project. SAS Forecasting for Desktop 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 for Desktop:

- 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 Project Manager for Desktop and using the SAS macros.

To view the token value for the user who is currently logged in, see the **Show my (*identity-token*) projects only** check box in the Projects dialog box.

If the project owner is set incorrectly, this value can be manually changed using the SAS Forecast Project Manager for Desktop client or the FSSETOWN macro. For more information, see "FSSETOWN Macro" on page 111.

Hard Crash Causes Projects to Remain Locked

If your system or client experiences a hard crash, projects that were open at the time of the crash might be left in a locked state.

When multiple sessions are accessing content, the processes must be synchronized to avoid data corruption. This synchronization involves keeping track of the locked state of open projects. In the SAS Forecasting for Desktop, the locked state is maintained in the non-editable file `!DESKTOPHOME\locks.dat`. Each running process creates a temporary file in ***!DESKTOPHOME***`\Processes` that indicates that the process is active. When the JVM exits normally (including if there is a soft crash), the temporary file is removed. Any associated locks in the `locks.dat` file are cleaned up when the next lock is requested. However, if the system suffers a hard crash, this process is compromised. You then must manually remove this temporary file in order for the software to detect that the process is no longer active. Projects that were open at the time of a crash might be corrupted due to the loss of the in-memory state. Hence, after the projects are unlocked, they should be deleted and safe copies restored from backup archives.

Troubleshooting the SAS Forecast Project Manager for Desktop

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.

In SAS Forecasting for Desktop deployments, this failure generally implies that you have more than one client open. Try closing all clients including SAS sessions using SAS Forecast Batch Interface macros and then attempt the action again.

Troubleshooting the Macros in the Batch Interface

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.
```

Index