



THE
POWER
TO KNOW.

SAS[®] Time Series Studio 13.1

Administrator's Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2013. *SAS® Time Series Studio 13.1: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Time Series Studio 13.1: Administrator's Guide

Copyright © 2013, SAS Institute Inc., Cary, NC, USA

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

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

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

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

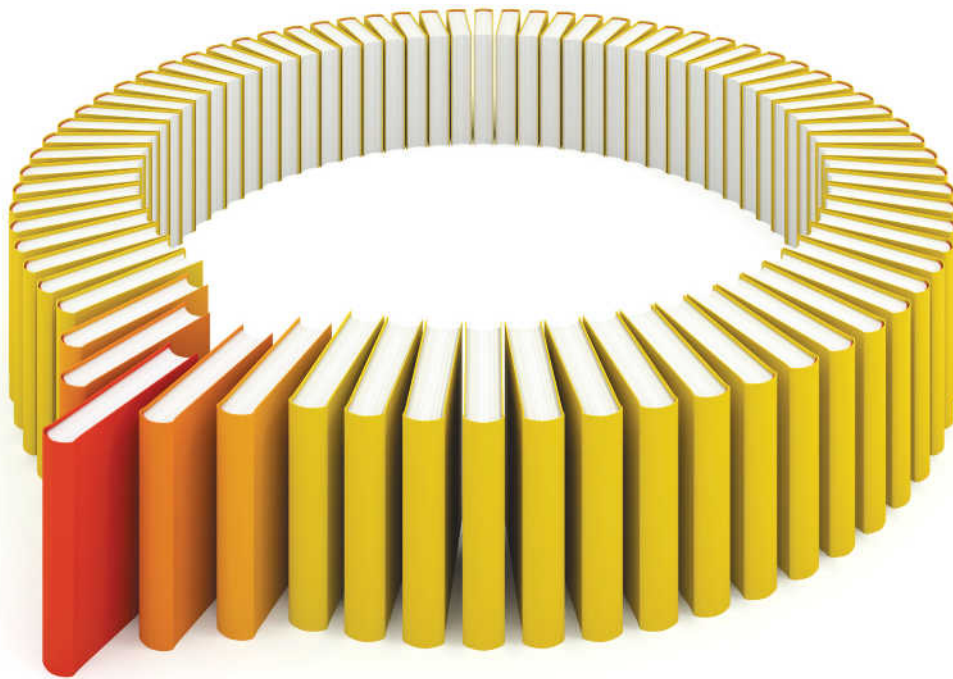
SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

December 2013

SAS provides a complete selection of books and electronic products to help customers use SAS® software to its fullest potential. For more information about our offerings, visit support.sas.com/bookstore or call 1-800-727-3228.

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

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



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

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

 support.sas.com/bookstore
for additional books and resources.


THE POWER TO KNOW.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. © 2013 SAS Institute Inc. All rights reserved. S107969US.0613

Contents

<i>Using This Book</i>	<i>vii</i>
<i>Recommended Reading</i>	<i>xi</i>

PART 1 Introduction to SAS Time Series Studio 1

Chapter 1 / Understanding SAS Time Series Studio	3
What Is SAS Time Series Studio?	3
Overview of the SAS Time Series Studio Architecture	3

PART 2 Installing SAS Time Series Studio 7

Chapter 2 / Pre-Installation Tasks	9
Verify System Requirements	9
Standard SAS User Accounts	9
Create an Operating System Account for Product Administrators and Users	12
Determine the Location of the SAS Environment URL	13
Chapter 3 / Installing SAS Time Series Studio	15
About the Deployment of SAS Time Series Studio	15
How to Install and Configure SAS Time Series Studio	15
Chapter 4 / Post-Installation Tasks	21
Minimum Requirements for Post-Installation	21
Create a Product Environment	21
Creating Operating System Accounts in UNIX Environments	22
Manage Roles and Capabilities	24

PART 3 Configuration Tasks 27

Chapter 5 / Specifying Security Permissions for Users and Groups	29
Enable Users to Access SAS Time Series Studio	30
Secure Access to SAS Time Series Studio	32
Chapter 6 / Creating and Configuring Libraries	37
Understanding Libraries	37
Enable Expanded Support for Libraries	39
How to Define a Pre-assigned Metadata Library	40
How to Define a Metadata Library That Is Assigned by SAS Time Series Studio ..	41

Create an Automatic Environment Library	42
Create a Configured Environment Library	42
Chapter 7 / Managing Product Environments	43
About Product Environments	43
Create a New Environment	44
Delete an Environment	45
Rename an Environment	45
View the Properties of an Environment	45
Register an Environment	46
Unregister an Environment	46
Chapter 8 / Managing Projects	47
Registering Projects	47
Archiving Projects	48
Chapter 9 / Miscellaneous Administration Tasks	51
Specify a Time-Out for SAS Time Series Studio	51
How to Specify the Value of the SAS Environment URL	52
Set the Preferred SAS Environment in the Logon Dialog Box for SAS Time Series Studio	52
Specify the Sample Size to Use in the Distribution Analysis	53
Remove a Configuration with the SAS Deployment Manager	53

PART 4 Using the SAS Time Series Studio Clients 55

Chapter 10 / SAS Time Series Studio	57
Requirements for Starting SAS Time Series Studio	57
Running SAS Time Series Studio from a Local Installation	57
Using Java Web Start	57
Adding Custom Time Intervals to SAS Time Series Studio	60
Chapter 11 / SAS Time Series Studio Batch Interface	65
Macros in the Batch Interface	65
Dictionary	67

PART 5 Appendixes 79

Appendix 1 / Troubleshooting SAS Time Series Studio	81
Gather Information	81
Additional Resources	84
Troubleshooting SAS Time Series Studio	86
Troubleshoot the Java Web Start Clients	88
Troubleshoot the Logon Process	89
Troubleshoot the SAS Time Series Studio Batch Interface	90

Using This Book

Audience

The *SAS Time Series Studio: Administrator's Guide* is intended for system administrators who need to install, configure, and optimize SAS Time Series Studio. SAS and other programming expertise are not required.

The administrator maintains the metadata for servers and other global resources that are required by SAS Time Series Studio.

Prerequisites

Review the system requirements documentation before you install the SAS Forecast Server to ensure that your system meets the requirements. For more information, see <http://support.sas.com/documentation/installcenter/en/ikforecastofrsr/65717/HTML/default/index.html>.

Accessibility

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

Recommended Reading

- *SAS Time Series Studio: User's Guide*
- Administration documentation for the SAS Intelligence Platform at <http://support.sas.com/documentation/onlinedoc/intellplatform>

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

SAS Books
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

Introduction to SAS Time Series Studio

Chapter 1

<i>Understanding SAS Time Series Studio</i>	3
---	----------

1

Understanding SAS Time Series Studio

<i>What Is SAS Time Series Studio?</i>	3
<i>Overview of the SAS Time Series Studio Architecture</i>	3
Architecture Diagram	3
The SAS Intelligence Platform Components	4
SAS Time Series Studio Components	5

What Is SAS Time Series Studio?

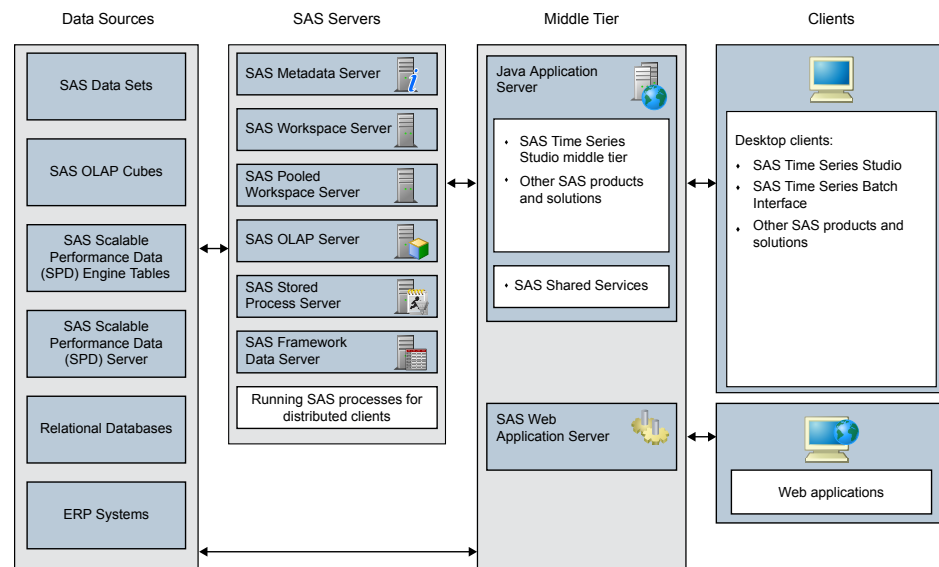
SAS Time Series Studio enables you to explore large volumes of timestamped or time series data. You can use this exploration for the purpose of data cleaning, time series aggregation, characterization, and segmentation of data. The data from SAS Time Series Studio can be exported for subsequent analysis such as forecasting, econometric analysis, pricing analysis, risk analysis, time series mining, and other analyses related to time series data

Overview of the SAS Time Series Studio Architecture

Architecture Diagram

The following figure shows how SAS Time Series Studio and the SAS Time Series Studio middle tier fit with the SAS Intelligence Platform.

Figure 1.1 Overview of SAS Time Series Studio Architecture



The SAS Intelligence Platform Components

SAS Time Series Studio works with the SAS Intelligence Platform, which is installed and configured when you deploy SAS Time Series Studio.

Integration Technologies

The SAS Metadata Server provides an enterprise-level repository for SAS server configurations and application management metadata. Products such as SAS Time Series Studio store metadata about users and other resources. Administrators use SAS Management Console to administer the SAS Metadata Server and the SAS server configurations. SAS Time Series Studio uses the SAS Metadata Server to obtain metadata about SAS libraries, the SAS Workspace Server, and the SAS Object Spawner. SAS Time Series Studio also uses the SAS Metadata Server to authenticate users.

The SAS Workspace Server provides all computation and intermediate data storage services. SAS Time Series Studio uses the SAS Workspace Server to save data to SAS data sets. SAS Time Series Studio accesses the SAS Workspace Server through the middle tier.

Web Infrastructure Platform

The SAS Intelligence Platform provides a service-oriented framework for building enterprise applications for the middle tier. Because the middle tier for SAS Time Series Studio is implemented using this framework, SAS Time Series Studio has core integration support for common infrastructure needs, such as auditing and resource publishing.

SAS Time Series Studio Components

SAS Time Series Studio consists of the following components:

SAS Time Series Studio Middle Tier

SAS Time Series Studio middle tier is deployed on an application server. The SAS Time Series Studio middle tier is a collection of custom services that are used to support the SAS Time Series Studio client.

SAS Time Series Studio client

SAS Time Series Studio is the primary client application. It enables you to interactively explore time series data. You can interactively structure the time series data from several hierarchical and frequency perspectives. You can interactively query (or subset) the time series data using hierarchical queries, graphical queries, parametric queries, or manual selection. Given a target series, you can interactively search for similar series. After retrieving time series data, you can analyze the data using common statistical time series analysis techniques. After you understand the patterns in the selected time series, the data can be exported for subsequent analysis such as forecasting, econometric analysis, pricing analysis, risk analysis, time series mining, and other analyses related to time series data.

SAS Time Series Studio Batch Interface

The SAS Time Series Studio batch interface consists of client macros to use with the SAS Foundation. These macros send requests to the SAS Time Series Studio middle tier to perform actions just like the SAS Time Series Studio client. Most actions focus on managing projects and environments.

Note: You should be aware that software orders of SAS Time Series Studio 13.1 also include SAS Forecast Server procedures. In the experimental release of SAS Time Series Studio 12.1, these procedures were called SAS High-Performance Forecasting.

Part 2

Installing SAS Time Series Studio

<i>Chapter 2</i>	
<i>Pre-Installation Tasks</i>	9
<i>Chapter 3</i>	
<i>Installing SAS Time Series Studio</i>	15
<i>Chapter 4</i>	
<i>Post-Installation Tasks</i>	21

2

Pre-Installation Tasks

Verify System Requirements	9
Standard SAS User Accounts	9
Overview of User Accounts	9
Internal User Accounts	10
Required External User Accounts	10
Rights and Permissions Required by External Accounts	11
Create an Operating System Account for Product Administrators and Users	12
About the User Accounts for SAS Time Series Studio	12
Create an Operating System Account for the Product Administrator	12
Create an Operating System Account for a SAS Time Series Studio User	13
Determine the Location of the SAS Environment URL	13

Verify System Requirements

Review the system requirements documentation before you install the SAS Forecast Server to ensure that your system meets the requirements. For more information, see <http://support.sas.com/documentation/installcenter/en/ikforecastofrsr/65717/HTML/default/index.html>.

Note: In order to launch SAS Time Series Studio via the Java Web Start client, you need Java Runtime Environment (JRE) 1.7 or later to be installed on the client machine.

Standard SAS User Accounts

Overview of User Accounts

You use two types of user accounts:

- Internal user accounts are accounts that are known only to SAS. They are created and authenticated in metadata internally rather than externally.
- External user accounts are accounts that are defined outside of SAS metadata. These accounts are local to a machine or are defined in a network directory

service of which the machine is a member, such as Lightweight Directory Access Protocol (LDAP).

Note: Before you run the SAS Deployment Wizard, the SAS Installer, SAS Spawned Server User, and SAS First User (if used) external user accounts must exist on each SAS Workspace Server that has SAS Time Series Studio projects.

Internal User Accounts

Internal user accounts are known only to SAS and are created and authenticated in metadata internally rather than externally. SAS identifies internal accounts by appending @saspw to the user ID. For two of the required user accounts, the SAS Administrator and the SAS Trusted User, the SAS Deployment Wizard prompts you to create internal user accounts.

The following table describes the default internal user accounts. (SAS internal accounts are authenticated on the metadata server.)

Internal User Account	User ID
SAS Administrator: The user account with privileges associated with the SAS Metadata Unrestricted Users role.	sasadm@saspw
SAS Trusted User: The user account that can impersonate other users on connections to the metadata server. Some SAS processes use this account to communicate with the metadata server on a client's behalf.	sastrust@saspw
SAS Anonymous Web User: The optional user account that can be used to grant web clients anonymous access to certain SAS Web Infrastructure Platform applications.	webanon@saspw

For more information about SAS internal user accounts, see [SAS 9.4 Intelligence Platform Installation and Configuration Guide](#).

Required External User Accounts

External user accounts are user accounts that are defined outside of SAS metadata. These accounts are local to a machine or are defined in a network directory service of which the machine is a member, such as LDAP. SAS requires external user accounts for two purposes: installing the software and running certain SAS server processes. You must define these user accounts before you run the SAS Deployment Wizard.

The following table describes the external user accounts for SAS Forecast Server.

External User Account	Recommended User ID
SAS Installer Used to install SAS. Do not use root as the SAS Installer user ID. (required)	sas

External User Account	Recommended User ID
SAS Spawnd Servers The process owner for stored process servers and pooled workspace servers. (required)	sassrv
SAS Demo User SAS First User (optional)	sasdemo

See [SAS 9.4 Intelligence Platform Installation and Configuration Guide](#) for more information about external user accounts in SAS.

Rights and Permissions Required by External Accounts

Operating systems require you to assign certain rights (or permissions) to the external user accounts that are used to deploy and run SAS.

External User Account	Required User Rights or Permissions
SAS Installer	The group that you designate as the primary group for the SAS Installer must contain the SAS Spawnd Servers account.
SAS Spawnd Servers account	Member of a group that is the primary group for the SAS Installer. (This group does not have to be the primary group for the SAS Spawnd Servers account.)

Note: Windows users must be granted **Log On as a Batch Job** user rights. In addition, the sassrv account must be granted **Act as part of the Operating System** user rights.

If you are running Windows, follow these steps to grant the **Act as part of the Operating System** user rights:

- From **Administrator Tools**, click **Local Security Policy**. Expand **Local Policies** and then click **User Rights Assignment**.
- In the Policy dialog box, right-click **Act as part of the operating system** and choose **Properties**. The Act as part of the operating system Properties dialog box appears. Click **Add User or Group**.
- The Select Users, Computers, Service Accounts, or Groups dialog box appears. In the **Enter the object names to select** window, type sassrv. Click **OK**. Click **OK** again.

Create an Operating System Account for Product Administrators and Users

About the User Accounts for SAS Time Series Studio

Users of SAS Time Series Studio must have a valid host operating system account, and you must associate that account with a metadata user. In addition, these users must have full access to the appropriate project areas on each SAS Workspace Server. User accounts can be created for SAS Time Series Studio as a pre-installation or post-installation task. For more information, see [“Create an Operating System Account for a SAS Time Series Studio User”](#) on page 13.

A product administrator account is a user account with additional permissions. The product administrator is not the same as a general administrator account, such as the SAS administrator (sasadm@saspw). You must create the operating system account for the product administrator as a pre-installation task. For more information, see [“Create an Operating System Account for the Product Administrator”](#) on page 12.

Create an Operating System Account for the Product Administrator

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

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

- Define the user (for example, `<domain>\username`) on the Microsoft Active Directory Server.
- If you are working on a local machine, complete these steps to create this user account:
 - 1 If you are running Windows 7, right-click the Computer icon on your desktop and select **Manage**. The Computer Management dialog box appears.
 - 2 In the left navigation pane, expand the **Local Users and Groups** node. The User and Group nodes appear.
 - 3 Right-click the **Users** node and select **New User**. The New User dialog box appears.
 - 4 In this dialog box, complete these tasks:
 - Specify a user name and password.

Note: In the Windows, you cannot enter `<domain>\username` (you enter the user name only), but you must enter `<domain>\username` in the SAS Deployment Wizard and SAS Management Console.

 - Clear the **User must change password at next logon** check box.
 - Select the **User cannot change password** check box.

- Select the **Password never expires** check box.

Click **Create**.

After you have created the user account, you must grant the permission **Log on as a Batch Job** in the local security settings.

Create an Operating System Account for a SAS Time Series Studio User

Create an operating system account for each SAS Time Series Studio user. For more information, see [“Creating Operating System Accounts in UNIX Environments” on page 22](#).

Determine the Location of the SAS Environment URL

During deployment, you are prompted by the SAS Deployment Wizard to specify a URL location of the SAS environment file, named `sas-environment.xml` (for example, `http://server-name:port/sas/sas-environment.xml`). This file defines a set of SAS deployments at your site for client applications to use. The `sas-environment.xml` file does not need to physically exist at the URL location that you specify in the SAS Deployment Wizard before beginning the SAS Installation. However, knowing the intended value of this URL is important because every client installation is prompted for this value. If you do not specify the URL when SAS Time Series Studio is installed, then as a post-installation task, you must manually edit a file on every client machine to specify this URL. Therefore, it is beneficial to decide on a value for this URL during your planning process so that it can be provided to administrators that might be performing an installation.

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

Note: If you include the HTTP server in your deployment, SAS automatically uses the URL for the packaged HTTP server.

3

Installing SAS Time Series Studio

<i>About the Deployment of SAS Time Series Studio</i>	15
<i>How to Install and Configure SAS Time Series Studio</i>	15

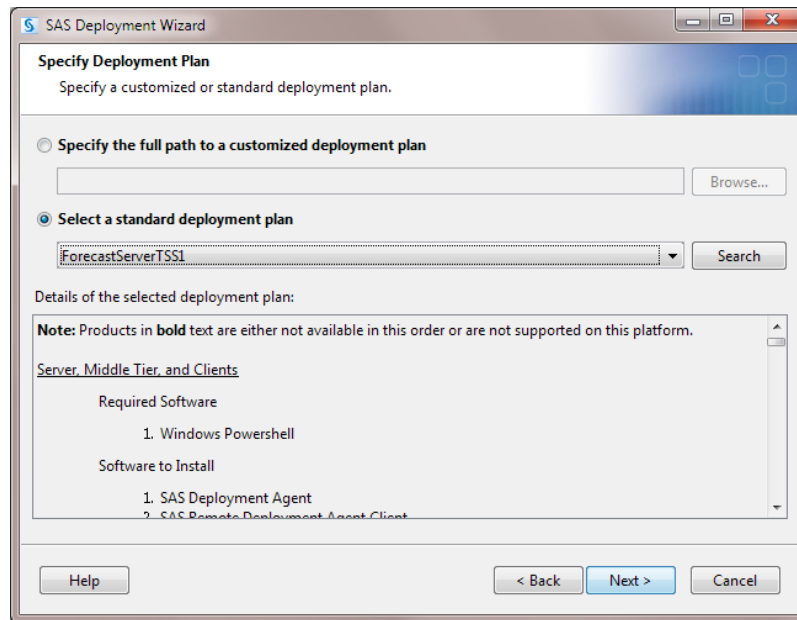
About the Deployment of SAS Time Series Studio

SAS Time Series Studio is a component of the SAS Forecast Server bundle, so when you install SAS Forecast Server, you can install SAS Time Series Studio at the same time. For more information about how to install SAS Forecast Server, see the *SAS Forecast Server: Administrator's Guide*. When you deploy SAS Time Series Studio, you deploy all of the components that are part of the SAS Time Series Studio architecture.

How to Install and Configure SAS Time Series Studio

To install and configure SAS Time Series Studio, you use the SAS Deployment Wizard. 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 Time Series Studio. Here is the information that you need to install SAS Time Series Studio:

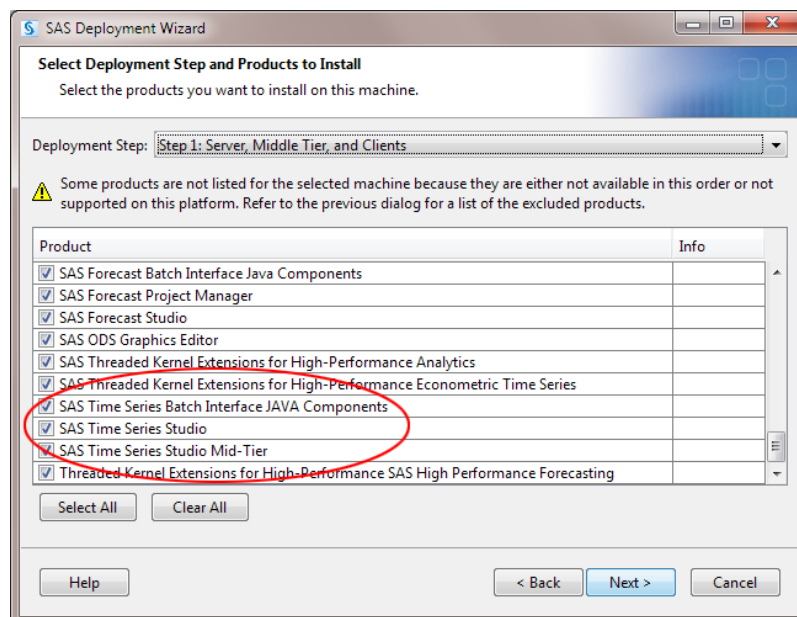
- 1 Start the SAS Deployment Wizard from your SAS Software Depot. For example, on a Windows system, double-click the setup.exe file, which is located in your SAS Software Depot folder.
- 2 In the Specify Deployment Plan step, select your deployment plan. It is recommended that you use one of the standard deployment plans created by SAS. If you need a custom plan, contact your SAS support personnel for assistance.



- 3 In the Select Deployment Step and Products to Install step, select the products that you want to install on this machine.

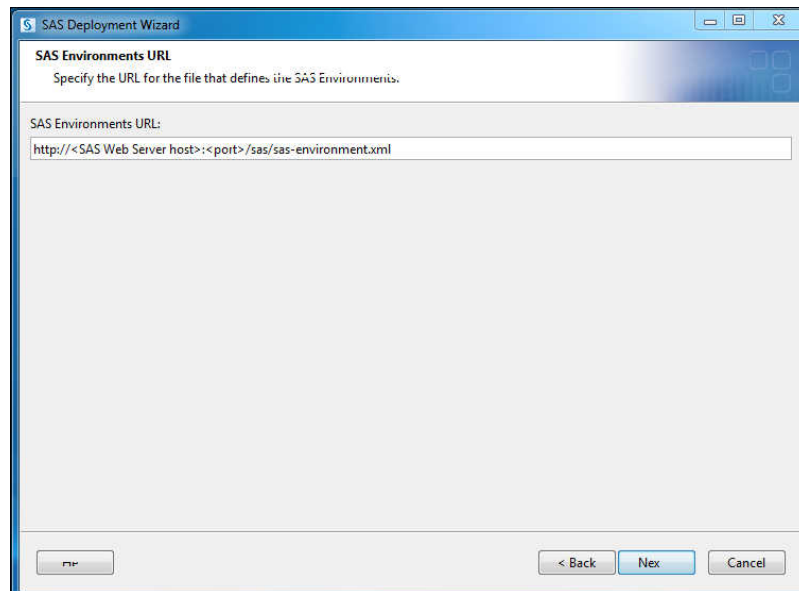
In this example, the following SAS Time Series Studio components are being installed on a single machine:

- SAS Time Series Studio Batch Interface Java Components
- SAS Time Series Studio
- SAS Time Series Studio Mid-Tier

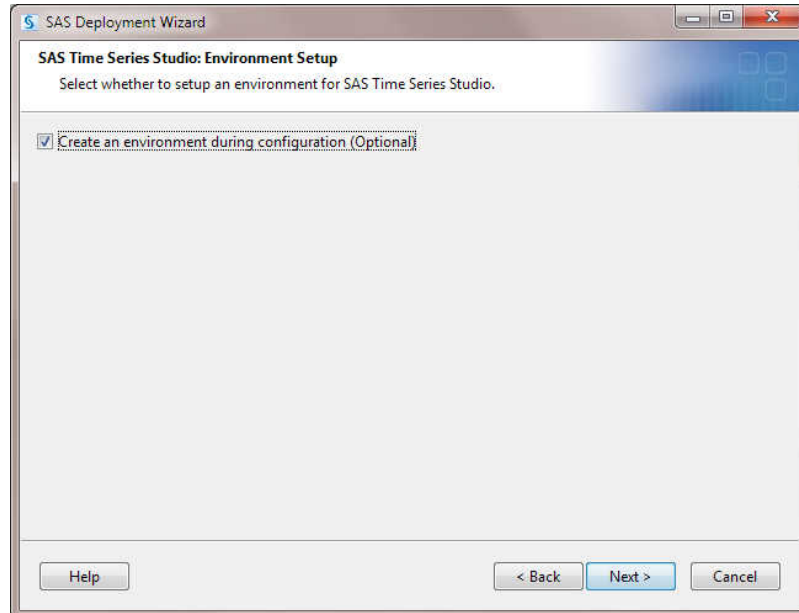


Click **Next**.

- 4 In the SAS Environments URL step, specify the URL location of the SAS environment file.



- 5 In the SAS Time Series Studio: Environment Setup step, you can select to **Create an environment during configuration**. Selecting this option enables you to create a product environment during configuration, not as a post-installation step. A product environment is required because it serves as a container for SAS Time Series Studio projects. The product environment should not be confused with a SAS environment. If you choose to create the product environment as a post-installation step, you can create it using SAS Time Series Studio. Click **Next**.



If you select the **Create an environment during configuration** check box, registration information for a product environment (named Default) is created.

The screenshot shows the 'SAS Deployment Wizard' window with the 'SAS Time Series Studio: Environment Setup' tab selected. The window contains three text input fields: 'Time Series Studio Environment Name' with the value 'Default', 'Environment Description' with the value 'Time Series Studio Default Environment', and 'Content Location' with the value 'C:\SAS\config\Lev1\AppData\SASTimeSeriesStudioMidTier12.3'. At the bottom, there are three buttons: 'Help', '< Back', and 'Next >', and a 'Cancel' button on the far right.

Note: The content location refers to a path on the SAS Workspace Server. The path that you specify for the content location must meet the requirements for a product environment. As a result, you might have to manually configure this path in SAS Time Series Studio as a post-installation step. For more information, see [“Managing Product Environments” on page 43](#).

In addition, if your server tier and middle tier are on different hosts and you are installing on the middle tier, then the default path that appears as the Content Location still refers to a path on the server tier. This path must exist on the server tier before the application is used.

- 6 In the SAS Internal Account: Time Series Studio Metadata User step, specify the password for the metadata user.

The screenshot shows the 'SAS Deployment Wizard' window with the 'SAS Internal Account: Time Series Studio Metadata User' tab selected. The window contains a text area with the instruction: 'Specify the metadata identity and the password for the Time Series Studio Metadata User Identity. The Metadata User is a privileged service account used by the SAS Time Series Studio to manage metadata objects.' Below this, there is a text area with the text: 'The Time Series Studio Metadata User is a SAS internal account stored in metadata.' There are five text input fields: 'Display Name' with the value 'Time Series Studio Metadata User', 'Name' with the value 'tssmeta', 'Internal User ID' with the value 'tssmeta@saspw', 'Internal Password' (empty), and 'Confirm Internal Password' (empty). At the bottom, there are three buttons: 'Help', '< Back', and 'Next >', and a 'Cancel' button on the far right.

- 7** In the Deployment Summary, review the products that you are about to install on your machine, and click **Start**.
- 8** 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

<i>Minimum Requirements for Post-Installation</i>	21
<i>Create a Product Environment</i>	21
<i>Creating Operating System Accounts in UNIX Environments</i>	22
Using Operating System Groups to Assign Permissions	22
Conditions for the SAS Time Series Studio User Group	22
Update the SAS Scripts to Grant Permissions to the SAS Time Series Studio User Group	22
<i>Manage Roles and Capabilities</i>	24
About Metadata Groups, Roles, and Capabilities	24
How to Configure SAS Time Series Studio Users	26

Minimum Requirements for Post-Installation

The number of post-installation and configuration tasks that you need to complete depends on your site. For example, your site might not use the Java Web Start client. If that is the case, you do not need to complete certain post-installation tasks.

Every site must complete the following steps to run SAS Time Series Studio:

- create a product environment
- set the permissions for each product environment

Create a Product Environment

When you install SAS Time Series Studio using the SAS Deployment Wizard, you have the option of creating a product environment. If you do not create a product environment during installation, then you must create the environment as a post-installation task. You can create this environment using SAS Time Series Studio. For more information, see [“Managing Product Environments” on page 43](#).

Creating Operating System Accounts in UNIX Environments

Using Operating System Groups to Assign Permissions

Users have different operating system privileges on the SAS Workspace Server. By defining a user group for SAS Time Series Studio, you can assign all SAS Time Series Studio users to the same group. Also, you can grant the same permissions to all SAS Time Series Studio users at one time. Each SAS Time Series Studio user must have Read, Write, and Execute permissions to each product environment directory that the user is permitted to use. A user also needs Read, Write, and Execute permissions to all of the files and directories in the product environment directory. The operating system must be configured to grant these permissions as new files and directories are created. The exact details of how this is done depends on which operating system groups are defined and your site's security policies.

Conditions for the SAS Time Series Studio User Group

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

- A group of SAS Time Series Studio users is created for the UNIX operating environment. The logon ID of each SAS Time Series Studio user must be in this group. The group must include any user who might run code that is created from a SAS Time Series Studio project in a SAS session.
- A user can be a member of multiple groups, but the SAS Time Series Studio user group is the primary group for each user.
- SAS scripts are updated to grant permissions to each SAS Time Series Studio user on the SAS Workspace Server. For more information, see [“Update the SAS Scripts to Grant Permissions to the SAS Time Series Studio User Group” on page 22](#).
- Each product environment directory has the correct ownership, and the SAS Time Series Studio user group members have Read, Write, and Execute permissions.

Update the SAS Scripts to Grant Permissions to the SAS Time Series Studio User Group

Using the UMASK option, you can grant permissions to a SAS Time Series Studio user on a conditional basis if the user is a member of the SAS Time Series Studio user group.

Note: The following example might require changes to match your server configuration. In particular, this example could result in changed permissions to other SAS files such as OLAP cubes. For example, if you are working with multiple UNIX user groups, and your site has a SAS OLAP Server, you must ensure that the

account under which the SAS OLAP Server runs maintains the Read and Execute permissions to OLAP cubes.

To grant permissions to the SAS Time Series Studio user group:

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

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

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

- a In `CMD=<your-operating-system-path>`, specify the full path on your SAS Workspace Server where the `id` command is stored. You can get this information by submitting a `which id` or `whence id` command.
- b In `GID=<solution-group-id>`, specify the group ID. Type `id` to get the GID and UID information.
- c A value of 002 is recommended for the `umask` option.

Here are code examples for each UNIX environment where SAS Time Series Studio is supported:

Operating Environment	Sample Code
AIX HP 64 (HP-RISC) HP 64 Itanium	<pre>CMD=/usr/bin/id CURR_GID=`eval \$CMD -g` GID=201 if [\$CURR_GID -eq \$GID]; then umask 002 fi</pre>
Solaris 64 Solaris X64	<pre>CMD=/usr/xpg4/bin/id CURR_GID=`eval \$CMD -g` GID=201 if [\$CURR_GID -eq \$GID]; then umask 002 fi</pre>
Linux	<pre>#!/bin/bash CMD=/usr/bin/id CURR_GID=`eval \$CMD -g` GID=500 if ["\$CURR_GID" -eq "\$GID"]; then umask 002 fi</pre>

Manage Roles and Capabilities

About Metadata Groups, Roles, and Capabilities

Default Roles for SAS Time Series Studio

SAS Time Series Studio provides default metadata roles. When SAS Time Series Studio is deployed, capabilities are already assigned to the default metadata roles. Using SAS Management Console, you can add capabilities to these default roles. You can create your own metadata roles and specify capabilities for those roles.

The following roles are provided with SAS Time Series Studio:

Role	Description
SAS Time Series Studio: Administrator	Enables the administrator to access the product, features, and all product content, regardless of ownership.
SAS Time Series Studio: User	Provides the user access to the product and features.

Capabilities in SAS Time Series Studio

The following table lists the capabilities in SAS Time Series Studio. Use these capabilities to prevent a user from administering the product.

Note: Granting a capability to a user does not necessarily ensure access to an action. Other settings (such as project configuration, project ownership, and metadata permissions) can limit access to an action.

Table 4.1 Capabilities for SAS Time Series Studio

Category	Capability	Description
Client Access	Time Series Studio	Enables the use of SAS Time Series Studio.

Category	Capability	Description
General Features	Administer Product	Enables the user to administer the metadata for and the projects in SAS Time Series Studio. Note: Only product administrators (users granted the Administer Product capability) can perform management actions and use the macros in the SAS Time Series Studio Batch Interface. Users who are assigned this capability are exempt from internal security checks, which are based on project ownership. However, these users are still subject to other restrictions imposed by other capabilities and metadata permissions.
	Legacy Features	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.
	Experimental Features	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.
	Debugging Features	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.

How to Configure SAS Time Series Studio Users

Note: On each SAS Workspace Server that contains product environments, you must have an operating system account for each SAS Time Series Studio user.

You can use the default metadata roles that are provided with SAS Time Series Studio to quickly configure your SAS Time Series Studio users. Each default metadata role is assigned capabilities that enable access to features.

To configure the metadata roles:

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

TIP To view the capabilities for a role, select the role, and click **Properties**.

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

Part 3

Configuration Tasks

Chapter 5	
<i>Specifying Security Permissions for Users and Groups</i>	29
Chapter 6	
<i>Creating and Configuring Libraries</i>	37
Chapter 7	
<i>Managing Product Environments</i>	43
Chapter 8	
<i>Managing Projects</i>	47
Chapter 9	
<i>Miscellaneous Administration Tasks</i>	51

5

Specifying Security Permissions for Users and Groups

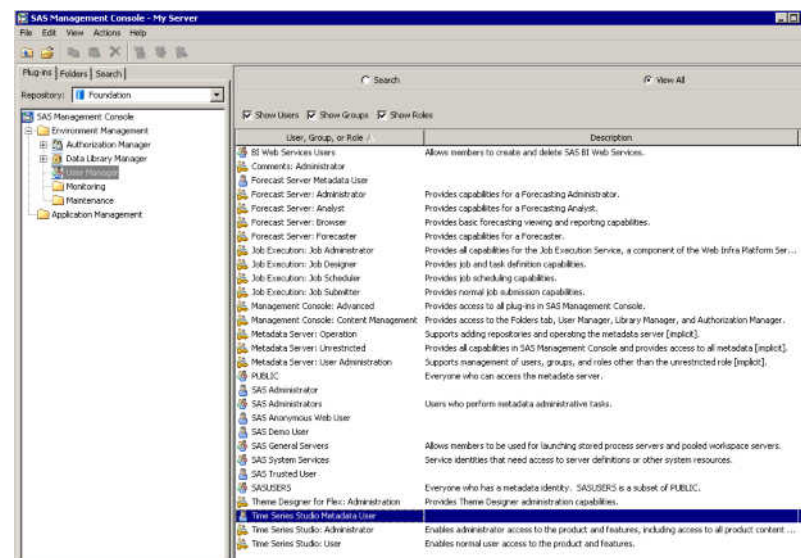
<i>Enable Users to Access SAS Time Series Studio</i>	30
Initial Users	30
The SAS Time Series Studio Metadata User	30
Project Owners	31
<i>Secure Access to SAS Time Series Studio</i>	32
Security Layers	32
Notes Regarding Permissions	32
Environment Permissions	33
Project Permissions	34
Sharing Options for Projects	35

Enable Users to Access SAS Time Series Studio

Initial Users

When you install and configure SAS Time Series Studio, standard user definitions and group definitions are added. The following display shows how these users and groups might appear in the User Manager plug-in for SAS Management Console.

Display 5.1 Example of Standard Users and Groups Created for SAS Time Series Studio



The SAS Time Series Studio Metadata User

What Is the SAS Time Series Studio Metadata User?

In SAS Time Series Studio, the Metadata User (tssmeta@saspw) has the capability to access data that is not accessible to the current user. The Metadata User can also update the metadata. The Metadata User account must have access to all metadata that is used by SAS Time Series Studio. For example, the WriteMetadata permission is required for any object that the product might need to modify, and the ReadMetadata permission is required for any object that the product might need to use or detect.

During deployment, the Metadata User is granted permissions to the `/System/Applications/SAS Time Series Studio` metadata folder. This folder contains the metadata objects that represent SAS Time Series Studio content.

If you create metadata folders outside of the `/System/Applications/SAS Time Series Studio` directory that you want to use with SAS Time Series Studio, you must grant the Metadata User the appropriate permissions to these folders. Special metadata objects that are used by SAS Time Series Studio (such as objects that

define libraries and servers) must be accessible. Permission settings must be set so that ReadMetadata permission is granted to all objects that are created during deployment.

Specify User Permissions for the Metadata User

To grant the Metadata User the appropriate permissions to a folder outside of the `/System/Applications/SAS Time Series Studio` directory:

- 1 Start SAS Management Console. Connect as a SAS administrator (for example, `sasadm@saspw`).
- 2 Click the **Folders** tab.
- 3 Locate the metadata folder that you created.
- 4 Right-click on the metadata folder, and then select **Properties**. The Properties dialog box appears.
- 5 Click the **Authorization** tab.
- 6 Next to **Users and Groups**, click **Add**. The Add Users and Groups dialog box appears.
- 7 In the **Available Identities** list, double-click **Time Series Studio Metadata User**. The Time Series Studio Metadata User is added to the **Selected Identities** list. Click **OK**.
- 8 In **Users and Groups**, select **Time Series Studio Metadata User**.
- 9 Grant the **ReadMetadata**, **WriteMetadata**, **WriteMemberMetadata**, and **CheckInMetadata** permissions to the Time Series Studio Metadata User, and then click **OK**.

TIP When appropriate, grant the ReadMetadata, WriteMetadata, WriteMemberMetadata, and CheckInMetadata permissions to the parent folder. A child folder automatically inherits these permissions from its parent folder.

Project Owners

What Is a Project Owner?

The user who creates a project in SAS Time Series Studio is the de facto administrator for that project. This user is referred to as the project owner. In addition to the default capabilities, a project owner can perform management actions, such as deleting a project.

The value for the project owner is detected by comparing a generated identifier for the user (called an identity token) with a previously stored token value for the user. To determine the previously stored token value for the user who is currently logged in, look at the session record on the SAS Time Series Studio Status page.

Note: In many cases, the identity token for a user matches the user name.

How to Change Project Ownership

You can change project ownership using the %TSSETOWN macro. For more information, see “%TSSETOWN Macro” on page 77.

Secure Access to SAS Time Series Studio

Security Layers

Security settings in SAS Time Series Studio are implemented in four layers.

- capabilities, which are enforced by SAS Time Series Studio
- metadata permissions
 - ReadMetadata, WriteMetadata, WriteMemberMetadata, and CheckInMetadata, which are enforced by the SAS Metadata Server
 - other permissions, such as library authorizations, which are enforced by SAS Time Series Studio
- project ownership and sharing, also called the application security model, which is enforced by SAS Time Series Studio
- file system permissions, which are enforced by the operating system

Typically, permissions are checked in the order in which they are listed. For example, capabilities are checked first and file system permissions are checked last. In order for a user to be able to access project content, all four layers of security must permit this access.

Notes Regarding Permissions

File System Permissions

Use metadata and SAS Time Series Studio permissions when you define your security strategy. Typically, file system permissions are not checked before an operation is attempted in a product, which could result in run-time errors.

Note: Only using file system permissions to secure content is not recommended.

WriteMetadata Permissions

In SAS Time Series Studio, the metadata security that is applied to an object with product content is limited to controlling the visibility of this content using the ReadMetadata permission. Metadata updates to an object are performed indirectly by the Metadata User. The WriteMetadata settings for an individual user account have no impact on the product's behavior.

What Permissions Can You Control Using Metadata?

To secure access to metadata objects that represent SAS Time Series Studio data, you can grant or deny permissions to a user or group using the **Authorization** tab in SAS Management Console.

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

Metadata Object	Permissions
environment	ReadMetadata—controls visibility
projects	ReadMetadata—controls visibility
libraries	ReadMetadata—controls visibility Read—controls ability to read contents Write—controls ability to write contents
tables that are registered to libraries	Read—controls ability to read contents Write—controls ability to write contents
columns in registered tables	Read—controls ability to read contents Write—controls ability to write contents

Note: Other permissions (such as using the ReadMemberMetadata permission to hide server objects) can impact the product's behavior. However, these other permissions are usually not needed under typical operating conditions.

Environment Permissions

About Environment Permissions

An environment is implemented using a metadata folder, a metadata object, and a directory structure (in the server file system). Only metadata and file system permissions apply to an environment. To enable project sharing within an environment, you must use the file system permissions to grant full control to all users who are allowed to access the environment. Metadata permissions should be used to secure the environment.

The ReadMetadata permission at the environment level controls whether a user can access the projects within an environment. When a user logs on to a SAS Time Series Studio client, the only environments that appear in the Projects dialog box are the environments where the user has ReadMetadata permission.

Note: Each file system location on a SAS Workspace Server should be used by only one environment system-wide, regardless of the number of environments or middle tiers in the system. In general, the file system location that is specified for the environment and its subdirectories should be reserved for that environment only. This restriction ensures the integrity of the file system's content and the ability to properly coordinate client access.

You can create environments by using the %TSNEWENV macro or the Environments dialog box in SAS Time Series Studio. For more information, see ["%TSNEWENV Macro" on page 75](#).

Configure Environment Permissions

To configure permissions for an environment:

- 1 In the file system, grant full control to all users and user groups who are allowed to access the environment. A user group for all users of the environment is recommended.
- 2 Start SAS Management Console. Connect as a SAS administrator (for example, sasadm@saspw).
- 3 Click the **Folders** tab.
- 4 Expand the **System**, **Applications**, **SAS Time Series Studio**, **Time Series Studio 13.1**, and **Environments** folders.
- 5 Right-click on an environment folder, and then select **Properties**. The environment Properties dialog box appears.
- 6 Click the **Authorization** tab.
- 7 Select a user or user group in **Users and Groups**.
- 8 Grant the **ReadMetadata** permission to the user or user group, and then click **OK**.

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

Project Permissions

About Project Permissions

A project is similar to an environment in physical structure. The content of a project is saved to an environment subdirectory. The security for the file system was specified when you created the environment. Because a project's content is saved in an environment subdirectory, the project files inherit the permissions from the environment. New project metadata is saved in the environment's metadata folder. The project inherits its permissions from the environment folder. As a result, no additional security is required when a project is created. However, in SAS Management Console, you can set the ReadMetadata permission on each project object to control that project's visibility to users.

Configure Project Permissions

To configure permissions for a project:

- 1 Start SAS Management Console. Connect as a SAS administrator (for example, sasadm@saspw).
- 2 Click the **Folders** tab.
- 3 Expand the **System**, **Applications**, **SAS Time Series Studio**, **Time Series Studio 13.1**, and **Environments** folders.
- 4 Select an environment folder.

- 5 Right-click on a project, and then select **Properties**. The project Properties dialog box appears.
- 6 Click the **Authorization** tab.
- 7 Select a user or user group in **Users and Groups**.
- 8 Grant the **ReadMetadata** permission to the user or user group, and then click **OK**.

Sharing Options for Projects

About Sharing Projects

By default, only the project owner or a SAS Time Series Studio administrator can open a project. By enabling project sharing, any user who can see a project can access the project.

Enable Project Sharing

To enable project sharing:

- 1 Open the Project Properties dialog box.
 - To open this dialog box from the Projects dialog box, click **Properties**.
 - If the project is currently open in SAS Time Series Studio, select **File ► Project Properties**.
- 2 Select the **Allow other users to view and edit this project** check box, and then click **OK**.

Sharing and Groups

Although project ownership and the sharing security model do not include the explicit concept of a user group, the use of environments does divide users into implied work groups. For example, all users who can access a particular environment can be considered a work group. Therefore, when you enable sharing of a project, you are essentially sharing the project with other members of the environment work group. A user who does not have access to the environment in which the project is stored cannot access the project. Use caution when copying projects that are shared into another environment because the work group changes based on the environment.

6

Creating and Configuring Libraries

<i>Understanding Libraries</i>	37
Overview of Libraries	37
External Library Assignments	37
Libraries Assigned by SAS Time Series Studio	38
Authorization Checks	39
<i>Enable Expanded Support for Libraries</i>	39
<i>How to Define a Pre-assigned Metadata Library</i>	40
<i>How to Define a Metadata Library That Is Assigned by SAS Time Series Studio</i>	41
<i>Create an Automatic Environment Library</i>	42
<i>Create a Configured Environment Library</i>	42

Understanding Libraries

Overview of Libraries

SAS Time Series Studio uses SAS libraries and data sets to manage and access project data. For more information about libraries, see the “SAS Libraries” chapter in *SAS Language Reference: Concepts* and the “LIBNAME Statement” topic in *SAS Statements: Reference*.

Note: Library names cannot begin with an underscore character. A library name that begins with an underscore is the naming convention reserved for an internal SAS library.

In SAS Time Series Studio, libraries are either assigned by SAS Time Series Studio or assigned by something external to SAS Time Series Studio. Library assignments that are made by a SAS system that is external to SAS Time Series Studio or a user are examples of external library assignments.

External Library Assignments

Libraries can be assigned when a SAS server session is created or in the start-up code for a specific product environment. Because these libraries are not assigned by SAS Time Series Studio, they are external library assignments.

Here are the four main sources of external library assignments:

- All SAS libraries that are provided with SAS are automatically assigned to each SAS session. Examples of Base SAS libraries are Sashelp, Sasuser, and Work.
- Libraries can be assigned by adding LIBNAME statements to your SAS Foundation configuration files or SAS server configuration files. The scope of the file (whether it is a SAS Foundation configuration file or a SAS server configuration file) determines the availability of the library.
- Libraries can be assigned by adding LIBNAME statements to the start-up code for a product environment. These libraries are available only for the associated environment. If you assign a library in the environment's start-up code, you must include a corresponding LIBNAME CLEAR statement in the code that shuts down the environment.
- Pre-assigned libraries that are defined in the metadata and associated with SAS Time Series Studio are automatically assigned when the server session is created.

Libraries Assigned by SAS Time Series Studio

SAS Time Series Studio enables you to manage and assign libraries on demand. Even if a library is not currently assigned, it appears in SAS Time Series Studio like any other library. When SAS Time Series Studio detects that a library is going to be accessed (for example, when a library is referenced), SAS Time Series Studio assigns the library before allowing access.

Because of security, on-demand libraries are not available in SAS Time Series Studio by default. SAS Time Series Studio has a configuration setting that enables the use of each type of on-demand library. Types include manually assigned metadata libraries, configured environment libraries, and automatic environment libraries.

Here are the three types of libraries that are assigned by SAS Time Series Studio:

- Libraries that are defined in the metadata, but they are not pre-assigned. When SAS Time Series Studio detects that one of these libraries needs to be used, the library is assigned using the META engine with the METAOUT=DATA option. This assignment method ensures that metadata permissions are followed when accessing the library.

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

- Configured environment libraries that are defined in the `<environment-directory>\Config\libs` data set. Each row of this data set represents a library assignment and contains the information required for the BASE engine:
 - ☐ the LIBNAME
 - ☐ the full path to the library directory
 - ☐ whether access to the library should be read-only

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

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

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

Configured and automatic environment libraries enable you to perform the following tasks:

- Manage libraries for users who have access to the file system.
- Define local libraries for individual environments.

Authorization Checks

A key to good library management and use is performing authorization checks on the data accesses that are attempted by users. How these authorization checks are performed depends on the type of library that the user is trying to access.

Two strategies are used.

- The SAS Intelligence Platform and the SAS Metadata Server require that a user has ReadMetadata permission on a library object for the library to be available. For libraries that are defined in the metadata, SAS Time Series Studio further requires that the user has Read permission for the user to have Read access to the contents of the library. It requires that the user has Write permission for the user to have Write access to the contents of the library. Permissions can also be set for elements within the library. An element might be a data set (also referred to as a table) or a data set variable (also referred to as a column). For example, when a client application tries to read a variable in a data set, SAS Time Series Studio searches for an associated Column, Table, or Library object in that order. An authorization check is performed on the first object that is found.

TIP You can create table objects for any type of metadata library using the Data Library Manager plug-in for SAS Management Console. For more information, see the online Help for the Data Library Manager plug-in for SAS Management Console.

- All other libraries are assigned using the BASE engine (for example, the configured and automatic environment libraries) or are assigned in a way that cannot be determined by SAS Time Series Studio.

Enable Expanded Support for Libraries

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

To enable this library support:

- 1 Start SAS Management Console. Connect as a SAS Administrator (for example, sasadm@saspw), and then connect to a metadata repository.

- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** node, and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the left pane, select **Time Series Studio Mid-Tier**.
- 6 In **Time Series Studio Mid-Tier > Library Access**, select whether users can use libraries using the **Manual-assign metadata libraries**, **Automatic environment libraries**, and **Configured environment libraries** menus.
- 7 Click **OK**.

How to Define a Pre-assigned Metadata Library

To define a pre-assigned metadata library:

- 1 Create a metadata definition for the library.
 - a Start SAS Management Console. Connect as a SAS Administrator (for example, sasadm@saspw), and then connect to a metadata repository.
 - b Expand the **Data Library Manager** node, and select the **Libraries** node.
 - c Right-click the library that you want to pre-assign, and select **Properties**. The *library-name* Properties dialog box appears.
 - d Select the **Options** tab.
 - e Click **Advanced Options**. The Advanced Options dialog box appears.
 - f Select the **Library is Pre-Assigned** check box. The selected library is assigned whenever a SAS session is started for one of the assigned SAS servers.
 - g Verify that the library is assigned to the SAS servers.
 - h Click **OK**.
- 2 Set the metadata permissions on the new library for the SAS Time Series Studio product administrator and users.
 - a In SAS Management Console, expand the **Data Library Manager** node, and select the **Libraries** node.
 - b Right-click the library, and select **Properties**. The *library-name* Properties dialog box appears.
 - c Select the **Authorization** tab.
 - d In **Users and Groups**, select the product administrator, and grant permissions by selecting the **Grant** check boxes. Select each SAS Time

Series Studio user, and grant permissions by selecting the **Grant** check boxes.

Verify that the tssmeta account has ReadMetadata permission on the library. If the tssmeta account does not have ReadMetadata permission, SAS Time Series Studio might not detect the library or might classify the library incorrectly.

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

How to Define a Metadata Library That Is Assigned by SAS Time Series Studio

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

- 1 Verify that SAS Time Series Studio is configured to support metadata libraries that are manually assigned.
- 2 Create a metadata definition for the library.
 - a Start SAS Management Console. Connect as a SAS Administrator (for example, sasadm@saspw), and then connect to a metadata repository.
 - b Expand the **Data Library Manager** node, and select the **Libraries** node.
 - c Right-click the library that you want to pre-assign, and select **Properties**. The *library-name* Properties dialog box appears.
 - d Select the **Options** tab.
 - e Click **Advanced Options**. The Advanced Options dialog box appears.
 - f Ensure that the library is assigned to the correct SAS servers.
 - g Click **OK**.
- 3 Set the metadata permissions on the new library object for the SAS Time Series Studio product administrator and users.
 - a In SAS Management Console, expand the **Data Library Manager** node, and select the **Libraries** node.
 - b Right-click the library, and select **Properties**. The *library-name* Properties dialog box appears.
 - c Select the **Authorization** tab.
 - d In **Users and Groups**, select the product administrator, and grant permissions by selecting the **Grant** check boxes. Select each SAS Time Series Studio user, and grant permissions by selecting the **Grant** check boxes.

Verify that the tssmeta account has ReadMetadata permission on the library. If the tssmeta account does not have ReadMetadata permission, SAS Time Series Studio might not detect the library or might classify the library incorrectly.

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

Create an Automatic Environment Library

To create an automatic environment library:

- 1 Verify that SAS Time Series Studio is configured to support automatic environment libraries.
- 2 In the `<environment-directory>\Libraries` directory, create a subdirectory for the automatic environment library. The name of the subdirectory implies the LIBNAME for the library assignment. For example, if the LIBNAME is Tssuser, then the name of the subdirectory should be `tssuser`.

Note: If the `Libraries` directory does not exist, then you need to create it. The name for this directory must follow SAS naming conventions.

- 3 Specify the security access for the library. By default, a library is assigned read-write access. If the library should have Read-Only access, append an `.r` to the directory name. For example, the `tssuser.r` directory creates the Tssuser library with Read-Only access.

When a user opens this environment in SAS Time Series Studio, the new library is available.

Create a Configured Environment Library

To create a configured environment library:

- 1 Verify that SAS Time Series Studio is configured to support configured environment libraries.
- 2 In the `<environment-directory>\Config` directory, create a `libs.sas7bdat` file. This data set should contain three character variables: LIBNAME, Path, and Read-only.

Note: If the Libs data set is not does not exist, SAS Time Series Studio tries to create it when a user opens this product environment in SAS Time Series Studio.

- 3 Edit the Libs data set to include the information for the new library. For example, to define the Tssuser library, you might use these values:
 - For the LIBNAME variable, specify `tssuser`.
 - For the Path variable, specify `C:\mylibs\tssuser`.
 - For the Read-only variable, specify `yes`.

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

7

Managing Product Environments

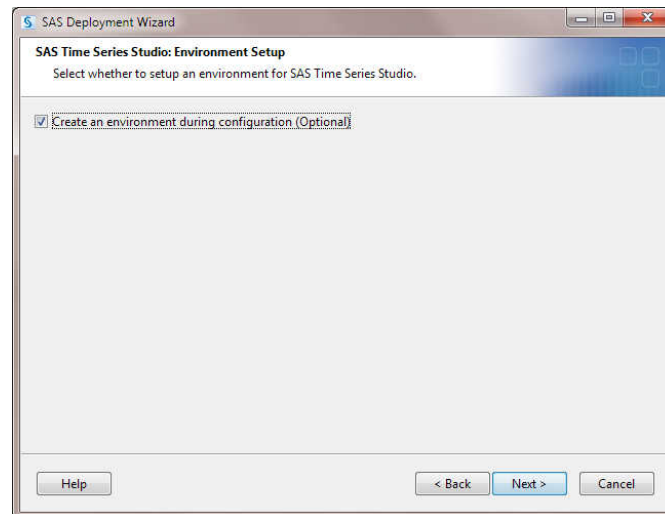
<i>About Product Environments</i>	43
<i>Create a New Environment</i>	44
<i>Delete an Environment</i>	45
<i>Rename an Environment</i>	45
<i>View the Properties of an Environment</i>	45
<i>Register an Environment</i>	46
<i>Unregister an Environment</i>	46

About Product Environments

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

A product environment is a product workspace for a product session. A product environment is created by the SAS Time Series Studio administrator, and is used only by SAS Time Series Studio. Product environments can be used to organize your projects.

You must create at least one product environment before you can create a project in SAS Time Series Studio. You can create a product environment when you install SAS Time Series Studio by using the SAS Time Series Studio: Environment Setup step in the SAS Deployment Wizard.



After SAS Time Series Studio is installed, you can create a product environment from the Projects dialog box. For more information, see [“Create a New Environment” on page 44](#).

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

- Your environments should not share the same file system location on the same server.
- You should not configure one environment to use a subdirectory within a different environment.
- Environment locations should not be shared across SAS deployments.

All product users who need to access the product environment must have full access to the file system content in that environment directory, including the root directory and any subdirectories. A product administrator needs to set permissions based on the specific operating environment. A product administrator should verify that every user has permission to access any new files that are created in the environment directory.

Create a New Environment

To create a new environment:

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.
- 3 Click **New**.

Delete an Environment

To delete an environment:

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.
- 3 Select the environment that you want to delete and click **Delete**.

Rename an Environment

To rename an environment:

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.
- 3 Select the environment that you want to rename and click **Rename**.

View the Properties of an Environment

To view the properties of an environment:

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.

- 3 Select the environment and click **Properties**. The Environment Properties dialog box appears.

Register an Environment

Registering an environment creates a new registration by using information from your file system content.

To register an environment:

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.
- 3 Click **Register**. The Register Environment dialog box appears
- 4 Enter a name for the environment.
- 5 (Optional) Type a description for the environment.
- 6 Select the server for the environment.
- 7 Specify the location of the environment in the file system.
- 8 (Optional) Select **Automatically register all projects within this environment** to register all of the projects in the environment. By default, projects are not registered when you register the environment.
- 9 Click **OK**.

Unregister an Environment

When you unregister an environment, no project content is deleted.

- 1 Open the Projects dialog box.
 - When you start SAS Time Series Studio, the Projects dialog box appears automatically.
 - After you close a project in SAS Time Series Studio, you can open the Projects dialog box by selecting **File ► Projects**.
- 2 Next to the **Environment** drop-down list, click **Manage**. The Environments dialog box appears.
- 3 Select the environment that you want to unregister and click **Unregister**.

8

Managing Projects

Registering Projects	47
Register a Project	47
Unregister a Project	47
Archiving Projects	48
Configure the Archive Functionality	48
Archive a Project	49
Unarchive a Project	49

Registering Projects

Register a Project

To register a project:

- 1 Select **File** ► **Projects**. The Projects dialog box appears.
- 2 Click **Advanced**.
- 3 Select the environment that contains the projects that you want to register.
- 4 Click **Register**. The Register Projects dialog box appears.
- 5 Select the projects to register and click **OK**.

Unregister a Project

To unregister a project:

- 1 Select **File** ► **Projects**. The Projects dialog box appears.
- 2 Click **Advanced**.
- 3 Select the environment that contains the projects that you want to unregister.
- 4 Select the project that you want to unregister and click **Unregister**. The Confirm Unregister dialog box appears.
- 5 Click **Yes** to unregister the project.

Archiving Projects

Configure the Archive Functionality

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

The archiving properties in SAS Time Series Studio enable you to control the process for creating and extracting project archives.

To modify archiving properties:

- 1 Start SAS Management Console. Connect as a SAS administrator (for example, sasadm@saspw).
- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** node, and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the left pane, select **Time Series Studio Mid-Tier**.
- 6 In **Time Series Studio Mid-Tier > Archiving**, you can set the following archiving properties:

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

- 7 Restart the SAS Time Series Studio middle tier for these changes to take effect.

Archive a Project

You can archive a project by using the Advanced Project Actions dialog box in SAS Time Series Studio or the %TSEXPORT macro.

To archive a project:

- 1 Select **File ► Projects**. The Projects dialog box appears.
- 2 Click **Advanced**. The Advanced Project Actions dialog box appears.
- 3 Select the environment that contains the projects that you want to archive.
- 4 Select the project that you want to archive and click **Archive**. The Archive dialog box appears.
- 5 Specify a name for the archive. By default, a name is provided. The name must be a valid SAS name, and it 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**.

Unarchive a Project

- 1 Select **File ► Projects**. The Projects dialog box appears.
- 2 Click **Advanced**. The Advanced Project Actions dialog box appears.
- 3 Select the environment where you want to unarchive the projects.
- 4 Click **Unarchive**. The Unarchive dialog box appears.
- 5 Select the files that you want to unarchive.
- 6 (Optional) Specify a name for each unarchived project. By default, the name of the project is the same as the original project name.
- 7 Click **OK**. The unarchived projects now appear in the Projects dialog box.

9

Miscellaneous Administration Tasks

<i>Specify a Time-Out for SAS Time Series Studio</i>	51
<i>How to Specify the Value of the SAS Environment URL</i>	52
<i>Set the Preferred SAS Environment in the Logon Dialog Box for SAS Time Series Studio</i>	52
<i>Specify the Sample Size to Use in the Distribution Analysis</i>	53
<i>Remove a Configuration with the SAS Deployment Manager</i>	53

Specify a Time-Out for SAS Time Series Studio

By default, a SAS Time Series Studio session times out after being idle for 60 minutes. SAS Time Series Studio checks every five minutes to see whether the SAS Time Series Studio session has timed out. You can change the values of the time-out and how frequently SAS Time Series Studio checks the session.

The time out is the maximum amount of time that should be allowed to elapse before SAS Time Series Studio assumes that the client session has failed. At this time, session resources are reclaimed, and any open resources (such as projects) might be corrupted. When specifying a time out, be sure to exceed the maximum time that any client would be idle under normal operations.

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

- 1 Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** node, and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the left pane, select **Time Series Studio Mid-Tier**.
- 6 In **Time Series Studio Mid-Tier > Session Timeout**, specify new values for the **Idle time limit** option and the **Timeout check interval** option.
Click **OK**.

- 7 Restart the SAS Time Series Studio middle tier for these changes to take effect.

How to Specify the Value of the SAS Environment URL

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

- 1 Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** folder, and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the left pane, select **Time Series Studio (Java Web Start)**.
- 6 In **SAS environment URL**, specify the URL for the sas-environment.xml file. Click **OK**.
- 7 Restart the SAS Time Series Studio middle tier for these changes to take effect.

Set the Preferred SAS Environment in the Logon Dialog Box for SAS Time Series Studio

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

Note: If the preferred SAS environment does not exist at run time, SAS Time Series Studio acts as if you did not specify a preferred environment.

To specify the list of SAS environments that should appear in the logon dialog box:

- 1 Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** folder, and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the left pane, select **Time Series Studio (Java Web Start)**.

- 6 In **Preferred SAS environment**, specify the name of the SAS environment that should be selected by default in the logon dialog box.
Note: Specify the value of the NAME= attribute in the sas-environment.xml file. This name is case sensitive.
- 7 If you do not want the user to be able to change the selected environment in the logon dialog box, set the **Lock SAS environment selection** property to `true`.
- 8 Click **OK**.
- 9 Restart the SAS Time Series Studio middle tier for these changes to take effect.

Specify the Sample Size to Use in the Distribution Analysis

If the input data source exceeds 100,000 observation, SAS Time Series Studio uses a random sample of the data. The default sample size is 10,000 observations.

You can specify a new sample size for SAS Time Series Studio. This sample size is used whenever the distribution option is selected in the client. In SAS Time Series Studio, distribution analyses are performed using the FREQ and UNIVARIATE procedures for the variables in the project.

To configure the sample size for distribution analysis in SAS Time Series Studio:

- 1 Before starting SAS Time Series Studio, open the tsxclient.ini file in a text editor. For example, in a default Windows installation, this file is located in the `c:\Program Files\SAS\SASTimeSeriesStudio\13.1` directory.
- 2 In the tsxclient.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, enter the following: `JavaArgs_12=-Dtss.distribution.sample.size=5000`

Remove a Configuration with the SAS Deployment Manager

To remove a configuration:

- 1 Navigate to the SASHome directory. For example, on a Windows system, navigate to `C:\Program Files\SASHome\SASDeploymentManager\9.4`.
Double-click `sasdm.exe` to launch the SAS Deployment Manager.
- 2 Select **Remove Existing Configuration**, and then click **Next**.
- 3 Select a configuration directory, and then click **Next**.
- 4 Specify your connection information to the SAS Metadata Server, and then click **Next**.

- 5 Select the product that you want to unconfigure, and then click **Next**. For example, to unconfigure SAS Time Series Studio, select **Time Series Studio 13.1**.
- 6 SAS Time Series Studio has metadata that describes user-defined environments and projects. This metadata is stored in the SAS Metadata Server. If you are removing a configuration for SAS Time Series Studio, you can also select to unregister the user content. This option removes the SAS Time Series Studio metadata when you remove the configuration. In either case, the project content in the file system is not deleted when you remove the configuration. You can use the register action to re-create the metadata later if desired.

Note: You can use the administrative functionality in SAS Time Series Studio to register and unregister projects. In addition, you can use the macros in the SAS Time Series Studio Batch Interface. For more information, see [“SAS Time Series Studio Batch Interface” on page 65](#).

- 7 In the **Summary** screen, click **Start**.



Part 4

Using the SAS Time Series Studio Clients

<i>Chapter 10</i>	
<i>SAS Time Series Studio</i>	57
<i>Chapter 11</i>	
<i>SAS Time Series Studio Batch Interface</i>	65

10

SAS Time Series Studio

<i>Requirements for Starting SAS Time Series Studio</i>	57
<i>Running SAS Time Series Studio from a Local Installation</i>	57
<i>Using Java Web Start</i>	57
How to Launch SAS Time Series Studio	57
Configure the Logon Behavior for Java Web Start	58
<i>Adding Custom Time Intervals to SAS Time Series Studio</i>	60
Create the Custom Time Interval	60
Example 1: StoreHours Custom Interval	61
Example 2: Creating a Custom Interval Definition Interactively	62

Requirements for Starting SAS Time Series Studio

Before you can start SAS Time Series Studio, you must start the servers (for example, the SAS Metadata Server and your web application server). For information about how to start these servers, see the *SAS Intelligence Platform: System Administration Guide*.

Running SAS Time Series Studio from a Local Installation

The SAS Time Series Studio client runs only on the Windows operating system. To start SAS Time Series Studio on the system where it is installed, select **Start ► All Programs ► SAS ► SAS Time Series Studio ► SAS Time Series Studio 13.1**.

Using Java Web Start

How to Launch SAS Time Series Studio

You do not need to install SAS Time Series Studio on every system where you want to run SAS Time Series Studio. Instead, you can launch SAS Time Series Studio by using Java Web Start.

To launch SAS Time Series Studio using Java Web Start, perform either of these steps:

- Open the SAS Time Series Studio Mid-Tier Current Status web page. The default URL for this page is the location where the SAS Time Series Studio middle tier is deployed. An example is `http://<your-server-name>:<port-number>/SASTimeSeriesStudioMidTier/Status`.

To start SAS Time Series Studio, click **Launch using Java Web Start**.

- Use the direct link to Java Web Start. You can launch SAS Time Series Studio from `http://your-server-name:8080/SASTimeSeriesStudioMidTier/main.jnlp`.

Configure the Logon Behavior for Java Web Start

Default Logon Behavior for the Desktop Application and Java Web Start

Each list of SAS environments has a default environment. When you log on to SAS Time Series Studio, the Log On dialog box displays the environment that was most recently used on that machine. If you are logging on to SAS Time Series Studio using the Java Web Start client, the list of SAS environments includes the **(host environment)** option. This option represents the deployment from which the Java Web Start client was downloaded.

Configure the Logon Behavior on the Client

On the client, you can configure the logon behavior of Java Web Start in either of these ways:

- You can specify the URL for the `sas-environment.xml` file in the `SAS_ENV_DEFINITION_LOCATION` environment variable.
- If you run the SAS Deployment Wizard on a client, and the URL for the `sas-environment.xml` file was specified during the SAS installation in the local `sassw.config` file, you can use the `SASHOME` environment variable to specify where the client deployment is installed. When the `SASHOME` environment variable is defined, the client automatically checks the configuration files of the current deployment for a configured `sas-environment.xml` file to use.

If you configure the logon behavior by using both ways, the value from the `SAS_ENV_DEFINITION_LOCATION` environment variable is used.

To create either of these environment variables on a client running Windows XP, perform the following steps. If you are running another version of the Windows operating system, then these steps might be slightly different.

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

Environment Variable	Variable Name	Variable Value
SAS_ENV_DEFINITION_LOCATION	SAS_ENV_DEFINITION_LOCATION	The location of the sas-environment.xml file. An example is <code>http://server-name:port/sas/sas-environment.xml</code> .
SASHOME	SASHOME	The location where SAS was installed. An example is <code>C:\Program Files\SASHome</code> .

Click **OK**. The SAS_ENV_DEFINITION_LOCATION and SASHOME variables now appear in the list of system variables.

4 Click **OK**.

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

Configure the Logon Behavior on the Server

Using the Configuration Manager in SAS Management Console, you can control the logon behavior for Java Web Start.

You can set the following options:

SAS environment URL

specifies the URL for the sas-environment.xml file. For more information, see [“Determine the Location of the SAS Environment URL” on page 13](#).

Preferred SAS environment

specifies the SAS environment that should be selected by default in the logon dialog box. For more information, see [“Set the Preferred SAS Environment in the Logon Dialog Box for SAS Time Series Studio” on page 52](#).

Lock SAS environment selection

specifies whether users can select a different SAS environment in the logon dialog box. When you specify both the **Preferred SAS environment** and **Lock SAS environment selection** options, you force users to use a specific SAS environment.

If you change any of these options, you must restart the SAS Time Series Studio middle tier or use a JMX call to apply the new values.

Adding Custom Time Intervals to SAS Time Series Studio

Create the Custom Time Interval

Although SAS Time Series Studio includes a variety of time intervals, your site might use custom time intervals.

To create a custom interval:

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

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

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

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

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

- 2 To define the name and location of your custom interval, specify the INTERVALDS= system option in either of these files:
 - in the config file (`sasv9_usermods.cfg` or `sasv9.cfg`) that is used by the SAS Workspace Server.
 - in the autoexec file (for example, `appserver_autoexec_usermods.sas`) that is used by the SAS Workspace Server.

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

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


The name for the *interval* must be 1–32 characters. The name cannot contain underscores or embedded numeric characters, except that the name can end with one or more digits. For example, StoreHours12 is a valid name, and Store12Hours is an invalid name. When you specify multiple intervals, each interval name must be unique. The value of the interval is the data set that is named in *libref.dataset-name*.

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

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

```
options intervals=(StoreHours=CustIntLib.StoreHoursDS)
```

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

Example 1: StoreHours Custom Interval

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

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

```
-intervals (StoreHours=CustIntLib.StoreHoursDS)
```

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

- a Assign the LIBNAME CustIntLib to the directory where the data set for the custom interval (in this example, StoreHoursDS) is stored.
- b Submit the following code to create the data set:

```
data CustIntLib.StoreHoursDS (keep=BEGIN END) ; 1
    start = '01JAN2012'D;
    stop  = '31DEC2012'D;
    do date = start to stop; 2
        dow = WEEKDAY(date);
        datetime=dhms(date,0,0,0);
        if dow not in (1,7) then
            do hour = 9 to 17;
                begin=intnx('hour',datetime,hour,'b');
                end=intnx('hour',datetime,hour,'e');
                output;
            end;
        else if dow = 7 then
            do hour = 9 to 13;
                begin=intnx('hour',datetime,hour,'b');
                end=intnx('hour',datetime,hour,'e');
                output;
            end;
    end;
```

```
format BEGIN END DATETIME.; 3
run;
```

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

The START= and STOP= options specify the date of the first and last observations in the data set.
- b The DO loop analyzes each observation from 01JAN2012 through 31DEC2012 (inclusive).
 - Use the WEEKDAY function to determine the day of the week for a specific date. By default, the days of the week are specified as 1 for Sunday, 2 for Monday, and so on. You need this information later in the DO loop to assign the correct store hours to the day of the week. For example, if the day of the week is not 1 (Sunday) or 7 (Saturday), then the store hours are 9 to 17. If the day of the week is 7 (Saturday), then the store hours are 9 to 13.
 - Use the DHMS function to return a SAS datetime value for the date.
 - Use the INTNX function to return the datetime value of the beginning of the interval that is *n* intervals from the interval that contains the given datetime value.
- c Finally, the FORMAT statement specifies a DATETIME. format for the values of the BEGIN and END variables.

Example 2: Creating a Custom Interval Definition Interactively

To create a custom interval definition interactively:

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

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

11

SAS Time Series Studio Batch Interface

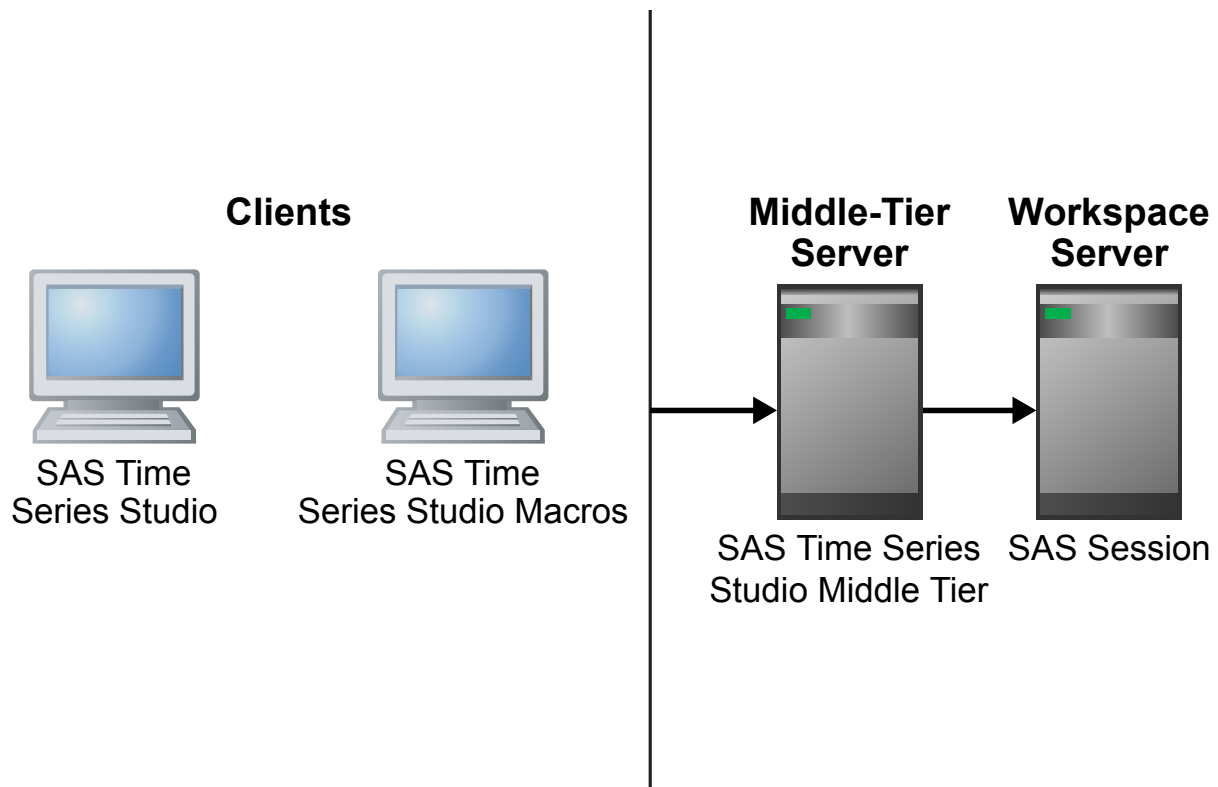
Macros in the Batch Interface	65
Summary of the Macros	65
Encrypting Your Passwords	67
Using Special Characters	67
Dictionary	67
%TSCOPY Macro	67
%TSEELARC Macro	69
%TSEELENV Macro	70
%TSEELPRJ Macro	71
%TSEEXPORT Macro	72
%TSIMPORT Macro	73
%TSLOGIN Macro	74
%TSLOGOUT Macro	75
%TSNEWENV Macro	75
%TSSETDEF Macro	76
%TSSETOWN Macro	77

Macros in the Batch Interface

Summary of the Macros

These macros provide a SAS language interface to the SAS Time Series Studio middle tier. (This interface is similar to the graphical user interface that SAS Time Series Studio provides to the middle tier.) Effectively, the macros are an alternative client to SAS Time Series Studio. Some macros enable you to create projects with features identical to a project that you created in SAS Time Series Studio. However, most of the macros focus on the management of environments and projects.

The following figure shows a schematic representation of how the macros interact with the other components in SAS Time Series Studio.



You can run these macros on any machine in your SAS Time Series Studio deployment that has an installation of SAS Foundation. On Windows, the macros reside in an autocall library in `!SASROOT\tsxbat\sasmacro`. On UNIX, they reside in the `SASFoundation/9.4/sasautos` directory.

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

Note: `%TSLOGIN` is the only macro that refers to SAS environments. The other macros in the SAS Time Series Studio Batch Interface use product environments.

The following table lists the macros in SAS Time Series Studio:

Macro Name	Description
<code>%TSCOPY</code>	Copies a project.
<code>%TSDELARC</code>	Deletes a project archive.
<code>%TSDELENV</code>	Deletes a product environment.
<code>%TSDELPRJ</code>	Deletes an existing project.

Macro Name	Description
%TSEXPORT	Exports a project to an archive.
%TSIMPORT	Imports a single project from an archive.
%TSLOGIN	Creates a new session for a specified instance of the middle tier.
%TSLOGOUT	Closes a session of a specified instance of the middle tier.
%TSNEWENV	Creates a new product environment.
%TSSETDEF	Stores a default value for a macro argument.
%TSSETOWN	Assigns an owner to the SAS Time Series Studio project.

Encrypting Your Passwords

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

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

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

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

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

Using Special Characters

Special characters, such as the apostrophe, are not allowed when you specify a directory path for a parameter. For example, if you specify an environment path of `C:\sas\TimeSeries'Studio\13.1\macro_projects` for the %TSNEWENV macro, the environment cannot be created, and an error message appears.

Dictionary

%TSCOPY Macro

The %TSCOPY macro copies a project to a new destination.

Syntax

%TSCOPY (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. Required arguments are separated by commas.

SOURCEENVIRONMENT = *environment-name*

specifies the name of the product 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.

DESTINATIONENVIRONMENT = *environment-name*

specifies the name of the product 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.

SOURCEPROJECTNAME = *source-project-name*

specifies the name of the SAS Time Series Studio 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.

DESTINATIONPROJECTNAME = *destination-project-name*

specifies the name of the SAS Time Series Studio 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.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

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, the directory for the archive folder and the remote archive folder are assumed to be 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.

REMOTEARCHIVEFOLDER = *directory-name*

specifies the directory in which the archived files of the source SAS Workspace Server can be found on the destination SAS Workspace Server (for example, `\\sourceserver\SAS\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 = *archive-name*

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

Results

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

&TSCOPY= SUCCESS | ERROR

Example

```
%tscopy(sourceprojectname=Shoe_Sales,
destinationprojectname=Shoe_Sales_2012,
sourceEnvironment=Default,
destinationEnvironment=Default
)
```

%TSDelARC Macro

The %TSDelARC macro deletes project archive.

Syntax

%TSDelARC (ARCHIVENAME= [, options])

Details

Required Arguments

You must specify the name of the archive.

ARCHIVENAME = *archive-filename*

specifies the name of the archived file that you want to delete.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

ENVIRONMENT = *environment-name*

specifies the name of the product environment. If the name is not specified, the currently stored value is used. The default is Default.

PROJECTNAME =*project-name*

specifies the name of the Time Series Studio project that is associated with the archive. This option is required only if you do not specify a value for the ARCHIVEFOLDER= option. If only project name is specified, then it is assumed that this project is registered with the specified environment. If both the project name and the archiveFolder are specified, then only the archiveFolder value is used.

ARCHIVEFOLDER = *folder-name*

specifies the name of the full path of the directory that contains the archived project. If not specified, the archive file is assumed to be in the working path of the specified environment.

Results

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

&TSDCLARC= SUCCESS | ERROR

Example

```
%tsdclarc(archivename=Shoe_Sales_2011)
```

%TSDCLENV Macro

The %TSDCLENV deletes a product environment.

Syntax

```
%TSDCLENV (ENVIRONMENT= [ , options ] )
```

Details

Required Arguments

You must specify the name of the environment.

ENVIRONMENT = *environment-name*

specifies the name of the product environment that you want to delete. This argument has no default value and must be explicitly specified.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

Results

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

&TSDelENV= SUCCESS | ERROR

Example

```
%tsdelenv(environment=testenv)
```

%TSDelPRJ Macro

The %TSDelPRJ macro deletes an existing project.

Syntax

%TSDelPRJ (PROJECTNAME= [, options])

Details

Required Arguments

You must specify a project name.

PROJECTNAME = *project-name*

specifies the name of the SAS Time Series Studio project. The value must be a valid SAS name. This argument has no default value and must be explicitly specified.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used.

If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

ENVIRONMENT = *environment-name*

specifies the name of the product environment. If the name is not specified, the currently stored value is used. The default is Default.

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

specifies whether the archives are to be deleted. The default is YES.

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

suppress errors if the project does not exist. The default is NO.

Results

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

&TSDelPRJ= SUCCESS | ERROR

Example

```
%tsdelprj (projectname=Shoe_Sales_2012,
environment=Default,
nowarn=NO
)
```

%TSEXPORT Macro

The %TSEXPORT macro exports a SAS Time Series Studio project to an archive.

Syntax

%TSEXPORT(ARCHIVENAME= [, options])

Details

Required Arguments

The following argument is required with %TSEXPORT.

ARCHIVENAME = *archive-name*

specifies the name of the archived file.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

ENVIRONMENT = *environment-name*

specifies the name of the product environment. If the name is not specified, the currently stored value is used. The default is Default.

PROJECTNAME = *project-name*

specifies the name of the project. If not specified, the currently stored value is used. The default name is Default.

DESCRIPTION = *description*

specifies a description to assign to the archive. The value is recorded in the Manifest.tss file in the archive. This file is available to all user interfaces that work with archive files.

ARCHIVEFOLDER = *folder-name*

specifies the folder where the archive is saved.

CPORT = YES | TRUE | NO | FALSE

specifies that data sets and catalogs are exported using cport. This is necessary if the project will be unarchived under a different platform. The default value is NO.

Results

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

&TSEXPORT= SUCCESS | ERROR

Example

```
%tsexport (archivename=Shoe_Sales_2011)
```

%TSIMPORT Macro

The %TSIMPORT macro imports archived files.

Syntax

%TSIMPORT(ARCHIVEPATH= [, options])

Details

Required Arguments

The following argument is required with %TSIMPORT.

ARCHIVEPATH = *archive-path*

specifies the path to the archive file.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not

specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

ENVIRONMENT = *environment-name*

specifies the name of the product environment. If the name is not specified, the currently stored value is used. The default is Default.

PROJECTNAME = *project-name*

specifies the name of the SAS Time Series Studio project. If not specified, the currently stored value is used. The default value is Default.

Results

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

&TSIMPORT= SUCCESS | ERROR

Example

```
%tsimport (archivepath=C:\SAS\TimeSeriesStudio\Projects\Archive\Sales_2012)
```

%TSLOGIN Macro

The %TSLOGIN macro creates a new middle-tier session on a specified instance.

Syntax

```
%TSLOGIN(USER=,PASSWORD= [, options])
```

Details

Required Arguments

The following arguments are required with the %TSLOGIN macro.

USER = *user-name*

specifies the user name to be used to log on to SAS Time Series Studio.

PASSWORD = *password-value*

specifies the user password to be used to log on to SAS Time Series Studio.

Options

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

SASENVIRONMENT = *environment-name*

specifies the name of the SAS environment containing the middle tier for SAS Time Series Studio. If not specified, SASENVIRONMENT=default is assumed. This argument is case sensitive.

MIDTIER = *midtier-identifier*

specifies the identifier associated with the instance of the middle tier to use. This is just a user-selected text label and would be important only when using multiple mid-tier instances within the same script. The default value is tssmain.

Results

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

&TSLOGIN= SUCCESS | ERROR

Example

```
%tslogin(user=name,
password=userpwd
)
```

%TSLOGOUT Macro

The %TSLOGOUT macro closes a middle-tier session on a specified instance.

Syntax

%TSLOGOUT([options])

Details**Options**

The following option can be used with the %TSLOGOUT macro.

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

Results

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

&TSLOGOUT= SUCCESS | ERROR

Example

```
%tslogout
```

%TSNEWENV Macro

The %TSNEWENV macro creates a new product environment. This call creates a skeleton for the environment.

Note: Creating a new product environment with the %TSNEWENV macro removes all existing files from the specified location.

Syntax

%TSNEWENV(ENVIRONMENT=,PATH= [, options])

Details

Required Arguments

The following arguments are required with the %TSNEWENV macro.

ENVIRONMENT = *environment-name*

specifies the name of the product environment. This argument has no default value and must be explicitly specified.

PATH = *path-name*

specifies the absolute file path to the base content directory.

Options

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

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

WSSERVER = *server-name*

specifies the name of the logical workspace server. This option is used only when creating environments for networked midtiers. The default is SASApp – Logical Workspace Server.

DESCRIPTION = *description*

specifies the description to be assigned to the environment.

Results

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

&TSNEWENV= SUCCESS | ERROR

Example

```
%tsnewenv(environment=Sales,
path=C:\SAS\TimeSeriesStudio\Environments\,
wsserver=Data_Server
)
```

%TSSETDEF Macro

The %TSSETDEF macro stores a defaults value for a macro argument..

Syntax

%TSSETDEF(MIDTIER=,KEY=,VALUE=)

Details

Required Arguments

The following arguments are required with %TSSETDEF.

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.

PROJECT specifies that the value is for a project variable.

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

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 the configurable default.

Options

There are no options for the %TSSETDEF macro.

Results

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

&TSSETDEF= SUCCESS | ERROR

Example

```
%tssetdef (key=environment,
value=envname,
midtier=*
)
```

%TSSETOWN Macro

The %TSSETDEF macro assigns an owner to a SAS Time Series Studio project.

Syntax

%TSSETOWN(OWNER= [, options])

Details

Required Arguments

The following argument is required with the %TSSETOWN macro.

OWNER = *user-name*

specifies the user name that owns the SAS Time Series Studio project.

Options

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

ENVIRONMENT = *environment-name*

specifies the name of the product environment. If the name is not specified, the currently stored value is used. The default is Default.

MIDTIER=*midtier-identifier*

specifies the identifier that is associated with the instance of the middle tier. This identifier is just a user-selected text label and is important only when using multiple instances of the middle tier within the same script. If this option is not specified, the middle tier that is associated with the most recent session is used. If an identifier does not exist, an error results. If you specify an asterisk (*) for this option, the currently stored value for the middle tier is used.

PROJECTNAME =*project-name*

specifies the name of the project. If not specified, the current stored default is used. The default is Default.

Results

The %TSSETOWN global macro variable indicates whether the %TSSETOWN macro finishes successfully or encounters errors:

&TSSETOWN= SUCCESS | ERROR

Example

```
%tssetown(owner=sastrust,
projectname=Sales_2012,
environment=Default
)
```



Part 5

Appendixes

Appendix 1

Troubleshooting SAS Time Series Studio **81**

Appendix 1

Troubleshooting SAS Time Series Studio

Gather Information	81
Overview	81
Operating System Environment and Configuration Information	82
Problem Description	82
Sample Data	83
Log Files	83
Additional Resources	84
The Status Page for the SAS Time Series Studio Middle Tier	84
JMX MBeans	85
Troubleshooting SAS Time Series Studio	86
Client Fails or Freezes during Start Up Due to Initial Environment Selection	86
Cluster Node Continues to Hold Locks after a Failure	87
Troubleshoot the Java Web Start Clients	88
Java Version Missing for Java Web Start	88
JNLP File Is Not Signed	88
Enable the Java Web Start Cache	88
Troubleshoot the Logon Process	89
Cannot Locate a SAS Environment URL during the Logon Process	89
Logon Process Cannot Locate Any Compatible SAS Environments	89
Troubleshoot the SAS Time Series Studio Batch Interface	90
CLASSPATH Variable Is Not Set	90
LOG4J System Property Is Not Initialized	90

Gather Information

Overview

When you are troubleshooting unexpected application behavior, it is important to isolate the problem and describe it and the context in which it occurs. There are general classes of information that can expedite resolving a technical problem:

- operating system environment 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 A1.1 Information Gathering Checklist

Information to Gather	Done
Details of your operating environment	[]
Detailed description of the problem (including what it takes to reproduce the problem)	[]
Sample data that would help reproduce the problem	[]
Log files	[]
Stack traces from any dialog boxes or consoles	[]

Operating System Environment and Configuration Information

If you request help from SAS Technical Support, then providing the following information about your installation can result in resolving the problem faster:

- Hardware platform, operating environment (including SAS version number), amount of physical memory, and number of processors.
- JDK version.
- JRE version.
- The SAS Time Series Studio version number and patch level.
- The configuration settings for the SAS Time Series Studio middle tier.
- Server language and locale.

Note: You must provide this information only once, unless it has changed from previous reports of this information.

Problem Description

Provide a problem description that includes as much information as possible. Include a description of the general task that you are trying to accomplish, your user ID, your roles and permissions, and what has happened during the SAS session. Provide answers to questions, such as the following:

- Are you working with new data or updating existing data?
- How easy is the problem to reproduce?
- What browser and version are you using?

- Is the problem specific to a locale? If so, which locales are having problems?

Sample Data

If possible, capture the data that caused the problem. SAS Technical Support might request your input data set so that it can better replicate your operating environment.

Log Files

SAS Time Series Studio Log

To view the SAS log in SAS Time Series Studio, select **Tools ► SAS Log**.

To include additional information in the SAS log:

- 1 Start SAS Management Console and connect as a SAS administrator (sasadm@saspw).
- 2 Expand the **Configuration Manager** and **SAS Application Infrastructure** nodes.
- 3 Right-click the **Time Series Studio Mid Tier 13.1** node and select **Properties**. The Time Series Studio Mid Tier 13.1 Properties dialog box appears.
- 4 Click the **Settings** tab.
- 5 In the selection pane, click **SAS Time Series Studio Mid-Tier**.
- 6 In the **Time Series Studio Mid-Tier > Logging** section, set these options:
 - Disable filtering. By default, messages about internal activities are omitted from the SAS log.
 - Enable the inclusion of JDBC messages. By default, JDBC messages are not included in the SAS log.

Click **OK**.

Java Stack Traces

When sending information to SAS Technical Support, send the complete full text of the Java stack trace. The logs for SAS Time Series Studio are in the standard location for the application container. An example of this path is `SAS_CONFIG\Web\Logs`. Log files for SAS Time Series Studio are in the form `SASTimeSeriesStudio*.log`. SAS Technical Support prefers that you do not send a screen shot of this information because often the screen shots do not include the full text of the trace.

Additional Resources

The Status Page for the SAS Time Series Studio Middle Tier

From the SAS Time Series Studio Mid-Tier Current Status page, you can see the current settings of the configuration options and you can monitor the activity of the current session. From this page, you can see the release numbers of the SAS Time Series Studio clients. This release information helps you determine whether the clients are compatible with the middle tier. The default URL for the status page is the location where the middle tier is deployed and appended with `/Status`. An example of this URL is `http://<your-server-name>:<port-number>/SASTimeSeriesStudioMidTier/Status`.

The SAS Time Series Studio Mid-Tier Current Status page provides the following information:

- A link from which you can launch SAS Time Series Studio as a Java Web Start client.
- General information, such as which version of SAS Time Series Studio you are running. This information is used by the SAS Time Series Studio clients to determine their compatibility with the middle tier.
- The configuration properties at run time of the middle tier.
- The configuration properties at run time of the Java Web Start client.
- The validation status of the SAS Workspace Servers, such as the SASApp - Logical Workspace Server, that are being used by SAS Time Series Studio.
- Information about the cluster node and resources that it is currently using. Note that an unclustered deployment is considered a cluster with only one node, so the information is present on this page even under deployments that are not using clustering.

Here is an example of a status page for the SAS Time Series Studio middle tier:

SAS® TIME SERIES STUDIO MID-TIER
Mid-Tier Node Status

ASSOCIATED COMPONENTS

SAS Time Series Studio [Launch using Java Web Start](#)

GENERAL INFORMATION

Mid-Tier Node ID	8ec5d89d-f6a1-4556-a86b-5674b7360ea3
Mid-Tier Location	RDCEX08046.race.sas.com:8080
Standard Build Information	View XML
Product Name	SAS Time Series Studio Mid Tier
Product Version	13.1
Product Build Name	sas.tsx.midtier.client.jar
Product Build Version	103000.0.0.20130812190000_d4tsx13
Product Build Time	0.0.20130812190000_d4tsx13
Configuration	Time Series Studio Mid Tier 13.1

ACTIVE CONFIGURATION

Session Timeout Limit (Minutes)	60
Timeout Polling Delay (Minutes)	5
Concurrent Session Requests Limit	0
Execute Environment Scripts	false
Execute Project Scripts	false
Allow Manual-Assign Metadata Libraries	false
Allow Automatic Environment Libraries	false
Allow Configured Environment Libraries	false
Safe Exception Name Patterns	java\.* javax\.* com\sas\analytics\ets\tsx.*

JMX MBeans

Although the status page for the SAS Time Series Studio middle tier provides a summary of the run-time state of the middle tier, you cannot modify the state from this page. The middle tier provides a simple JMX MBean that you can use to modify the state of the middle tier. For some modifications, you can use JMX calls to apply changes to the middle tier without restarting it. The JMX MBean provides details about the run-time state, but these details are limited to information directly related to actions that are supported by the JMX MBean.

Here are the actions supported by the JMX MBean:

- General Operation
 - ☐ `isInitialized()` reports whether the middle tier has completed the initialization that is performed at start-up.

- `reloadConfiguration()` reloads the configuration of the middle tier from the metadata and attempts to apply any changes to it.

Note: Configuration information is also cached in the Configuration Service. The Configuration Service must be reloaded before the `reloadConfiguration()` call to avoid getting the old configuration information from cache. The Configuration Service provides a JMX call that can be used to reload the cached information without restarting the middle tier for the Web Infrastructure Platform.

■ Server Validation

- `getServerStatus()` returns the validation information for all tested SAS Workspace Servers.
- `refreshServer(<server-name>)` clears the stored validation information for the specified SAS Workspace Server. Clearing this information results in the SAS Workspace Server being retested when it is accessed again. You perform this action when a server has been classified as unusable, but you have corrected the problem.
- `refreshAllServers()` clears the stored validation information for all SAS Workspace Servers.

■ Product Session Management

- `getSessions()` reports all active sessions in the SAS Time Series Studio middle tier.
- `killSession(<ID>)` forces the specified product session to close. (This action does not close the associated SAS login session.)

Note: Only use this action to close product sessions that are no longer associated with clients. For example, you might use this action when a client process fails and must be manually terminated. Forcing a session on the product middle tier to close causes any active client to fail. This action could result in data corruption, depending on what task the active client was performing when the product session was closed.

- `removeProcessMarker(ID)` removes the indicator that indicates that the cluster node is still running. You can use this to signal that the cluster node associated with the given ID has failed. This signal ensures that the remaining nodes will recover any orphaned resources. For example, any stale locks are removed.).

TIP After executing a JMX call, you can confirm your modification by using the status page for the SAS Time Series Studio middle tier.

Troubleshooting SAS Time Series Studio

Client Fails or Freezes during Start Up Due to Initial Environment Selection

In order for the initialization process to be performed, clients of SAS Time Series Studio must select a working product environment during start-up. However, the

initialization process occurs in the software before the user is granted control, and as a consequence, the software arbitrarily selects a product environment.

There are two known cases where this behavior can cause a failure during start-up:

- The selected environment has an AuthDomain for which no stored credentials are available. In this scenario, the initialization process executes before the credentials challenge prompt is configured. As a result, rather than the expected run-time challenge prompt, a run-time error can occur when no valid credentials are available.

Note: Due to architectural limitations, run-time challenge prompts are supported only by some of the product clients. Some clients can fail similarly without a challenge prompt even after initialization due to not supporting run-time challenges. In those cases, credentials must be stored to enable the proper use of that environment with the failing client.

- The selected environment is improperly configured (for example, uses an invalid directory path). In this scenario, the unexpected failure that results from the misconfiguration can prevent the initialization process from completing normally.

Note: Management actions to correct or remove misconfigured environments can also cause failures, depending on the nature of the configuration error. If you encounter an environment that does not work properly with normal management operations, contact SAS Technical Support for assistance.

In either case, to avoid problems like those above, the user can use the product clients to control the working environment that is used during initialization. Add `tss.environment.default=environment-name` to the VM arguments for the client to set the initial working environment.

Cluster Node Continues to Hold Locks after a Failure

In a clustered environment, if a cluster node experiences a failure, then any environments and projects that are being accessed through that node might be left in a locked state.

When multiple nodes are accessing content, the processes must be synchronized to coordinate who is currently accessing data at any point in time. For example, if a session on a particular node is currently running the Pricedata project, then the sessions on the other cluster nodes should not be allowed to access that project at the same time. This is to avoid data corruption, which can occur if multiple sessions access the same resource simultaneously.

Access to content must be synchronized both within cluster nodes and across cluster nodes. To accomplish the latter, each resource has a “lock” marker that shows which (if any) cluster node currently has a lock on that resource. There is also a process marker that indicates that a given cluster node is still running. When the cluster node process terminates normally, its process marker is removed and all locks are released. However, in the event that a node process crashes, the “lock” and “process” markers are not removed and continue to block access to the associated resources. In this case, you must manually remove the process marker to signal that the failed node process is no longer running.

In order to manually remove a marker, you must know the identifier for the failed cluster process. In a clustered environment, each cluster node has its own individual Status page. A Mid-Tier Process Coordination Status section has been included on the Status page for each node. This section includes information about any locks that are currently held by cluster processes. The identifier for each node is also listed

in this section. Usually, this information is sufficient to identify the failed process, based on identifying the resources that are inaccessible. With the Mid-Tier Node ID from this table, you can invoke the JMX action `removeProcessMarker` to remove the process indicator. After this is done, the remaining cluster processes handle the remaining cleanup work, including unlocking the resources.

In this example, the process marker is 7d158a86-4a3e-46fd-b036-9beb155e4976.

MID-TIER PROCESS COORDINATION STATUS	
Mid-Tier Process	Active Resources
 Mid-Tier Node ID	 Currently Held Lock
 7d158a86-4a3e-46fd-b036-9beb155e4976	 Default:Price: data
	 Default:[READER]

For more information about JMX, see “Additional Resources” on page 84.

Troubleshoot the Java Web Start Clients

Java Version Missing for Java Web Start

Java Web Start does not work because the required version of the JRE is not installed on the client tier. For SAS 9.4 products, the required Java version is 1.7 or later. To download the JRE, see <http://support.sas.com/resources/thirdpartysupport/v94>.

JNLP File Is Not Signed

When you launch the Java Web Start client, you get a warning that the JNLP file is not signed. For SAS Time Series Studio, a JNLP file is dynamically generated. It cannot be signed. You can ignore this warning.

Enable the Java Web Start Cache

The Java Web Start client does not work properly if the Java Web Start cache is disabled. By default, the cache is enabled. If the cache is disabled for any reason, complete these steps to enable it:

- 1 On Windows 7, select **Start** ► **Settings** ► **Control Panel** ► **Programs** ► **Java**.
- 2 Click the **General** tab.
- 3 In the **Temporary Internet Files** section, click **Settings**, and select the **Keep temporary files on my computer** check box.

Troubleshoot the Logon Process

Cannot Locate a SAS Environment URL during the Logon Process

During the logon process, SAS Time Series Studio clients look for a URL that references a `sas-environment.xml` file. This file contains the list of SAS environments that a user can select during the logon process. The logon process searches the following locations in order of precedence:

- 1 The `sas.env.definition.location` (a Java system property).
- 2 The `env.definition.location` (a Java system property).
- 3 The `SAS_ENV_DEFINITION_LOCATION` environment variable for the operating system.
- 4 The configured URL that is stored in the `SASENVIRONMENTSURL` property in `%{sas.home}/sassw.config`. (`sas.home` is a Java system property.)
- 5 The configured URL that is stored in the `SASENVIRONMENTSURL` property in `$SASHOME/sassw.config`. (`SASHOME` is an environment variable for the operating system.)

During the local deployment of any SAS Time Series Studio client, you are prompted for the URL value to store in the `sassw.config` file. If the SAS Time Series Studio client cannot locate this URL, you should verify that the information in the `sassw.config` file is correct.

Launching a SAS Time Series Studio client using Java Web Start does not require this URL to be configured. However, if the URL has been configured on either the client machine or in the configuration properties for Java Web Start, the SAS Time Series Studio client might generate an error if the configured URL cannot be used. To resolve this error, verify that a valid URL is specified. You can choose to remove the URL because it is optional for the Java Web Start clients.

Logon Process Cannot Locate Any Compatible SAS Environments

During the logon process, SAS Time Series Studio runs a compatibility test on each SAS environment. This test is primarily used to ensure that each SAS environment that a user can select in the logon dialog box is associated with a compatible deployment of SAS Time Series Studio. Only SAS environments that pass the compatibility test appear in the logon dialog box.

If no SAS environments pass the test, an error message states that no compatible SAS environments could be found. If you get this error message, verify the following requirements:

- The `sas-environment.xml` file that you are using includes at least one SAS environment with a SAS Time Series Studio deployment.
- The version of the SAS Time Series Studio deployment matches the version of the client. The compatibility test uses the build version of `sas.tsx.midtier.client.jar`.

Each SAS Time Series Studio deployment lists its version number on its middle-tier status page. The corresponding client version can be determined from the client picklist.

- The servers associated with the deployment are running and responsive.

Troubleshoot the SAS Time Series Studio Batch Interface

CLASSPATH Variable Is Not Set

You do not have to set the CLASSPATH variable before using the macros. If you get the following message while executing the macros, you can 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 get the following warning message while executing the macros, you can ignore it:

```
log4j:WARN No appenders could be found for logger java-class-name
log4j:WARN Please initialize the log4j system properly.
```